Allocating Costs in a Collaborative Transportation Procurement Network

Okan Örsan Özener
Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, oozener@isye.gatech.edu

Özlem Ergun
Industrial and Systems Engineering and The Logistics Institute, Georgia Institute of Technology, Atlanta, Georgia 30332, oergun@isye.gatech.edu

We study a logistics network where shippers collaborate and bundle their shipment requests in order to negotiate better rates with a common carrier. In this setting, shippers are able to identify collaborative routes with decreased overall empty truck movements. After the optimal routes that minimize total cost of covering all the shippers’ demand are determined, this cost is allocated among the shippers. Our goal is to devise cost allocation mechanisms that ensure the sustainability of the collaboration. We first develop cost allocation mechanisms with well-known properties from the cooperative game theory literature, such as budget balance, stability and cross monotonicity. Next, we define a set of new properties, such as a guaranteed discount from the stand alone cost for each shipper, desirable in our setting and propose several cost allocation schemes that could lead to implementable solutions. We also perform a computational study on randomly generated and real-life data to derive insights on the performance of the developed allocation schemes.

Key words: collaboration, cost allocation, truckload shipping.

History:
based on its existing lane network and the lanes it is anticipating to get by the time of service. The shipper procures the transportation service from the carrier that offers the lowest price for the shipper’s freight request. A key aspect that affects the carrier’s operational cost is asset repositioning. Asset repositioning, equivalently deadheading, is an empty truck movement from a delivery location to a pickup location. Carriers often have to reposition their assets to satisfy the demands of different shippers. Asset repositioning decreases the capacity utilization of the carrier, which results in an increase in operational costs. According to the estimates of American Trucking Association, the ratio of empty mileage to total mileage for Large Truckload Carriers is approximately 17% whereas the same ratio is approximately 22% for Small Truckload Carriers, which together corresponds to approximately 35 million empty miles monthly (ATA 2005). Since these deadhead miles generally result from the imbalance of freight requests of different shippers, asset repositioning costs are reflected in the lane prices.

A collaboration for procurement of transportation services is established when shippers get together to minimize their total transportation costs by better utilizing the truck capacity of a carrier. Nistevo, Transplace, and One Network Enterprises are examples of collaborative logistics networks that enable shippers and carriers to manage their transportation activities. A variety of companies such as General Mills, Georgia-Pacific, and Land O’Lakes are able to identify routes with less asset repositioning for their transportation needs, which result in considerable savings in their transportation expenses by being members of such collaborative logistics networks. For example, after forming collaborative partnerships with others in the Nistevo Network, Georgia-Pacific’s percentage of empty movements decreased from 18% to 3%, which corresponds to $11,250,000 savings yearly (Strozniak 2003).

The collaborative setting we consider in this paper is relevant for companies that regularly send truckload shipments, say several days of the week, and are looking for collaborative partners in similar situations to cross utilize a dedicated fleet. The truckload shipments that participate in such a collaboration are executed periodically. Since dedicated fleets are used for such shipments and the trucks are expected to return back to the initial location at the end of each period, the shippers are responsible for the anticipated asset repositioning cost. In practice, different carriers may have different costing schemes associated with empty truck movements. For instance, the anticipated asset repositioning cost along a lane can be a percentage of the lane cost and this percentage may be different for each lane of the network. The carrier may also charge a fixed amount for asset repositioning. Initially, we assume that the carrier charges a specific fraction of the original lane
cost for each asset repositioning. We call this fraction the “deadhead coefficient” ($\theta$). We also show that our methodologies are valid for other commonly used pricing structures.

The total asset repositioning costs incurred by a carrier depend on the entire set of lanes served by the carrier at the time of service. However, when a carrier gives a price to a shipper lane neither the carrier nor the shipper has perfect information on the final lane network the carrier will cover at the time of service. Hence, the price the carrier offers to a shipper includes a mark-up due to the expected repositioning cost associated with the shipper’s lane. On the other hand, if the shipper is able to bundle its lanes with complementary ones and provide a continuous move with minimal repositioning at the time of purchase, then the shipper will be able to negotiate for better rates from the carrier.

Although the benefits of forming collaborations are appealing, ensuring the sustainability of a collaboration is the key for realizing these benefits. A successful membership mechanism should distribute the joint benefits/costs of collaborating among the members of the collaboration in a “stable manner” (which will be presented later in the paper). Current practice in truckload transportation markets is to allocate joint costs proportionally to the cost of servicing the participating shipper’s lanes before collaboration. In this study, we first show that although such proportional allocation schemes are easy to implement, they are not stable from a game theoretic point of view. Furthermore, we demonstrate that the potential benefits from a shippers’ collaboration depend on the synergies among the lanes, hence, even a small modification in the lane network may cause the cost allocation to each shipper to change significantly. We then attempt to design cost allocation mechanisms which are stable, encourage the expansion of the collaboration, and guarantee a reduction over the base cost to each shipper.

Before illustrating the concept of collaboration on this problem and the challenges associated with this collaborative setting, we first introduce some terminology. Let “original lane cost” be the cost of performing a full truckload delivery from the lane’s origin to the lane’s destination. Then, the “asset repositioning cost” along the an arc is equal to $\theta$ times the original lane cost. “Stand alone cost of a set of lanes” is the minimum cost of serving these lanes (the original lane costs plus the asset repositioning costs). Similarly, “stand alone cost of a shipper” is the total cost of covering all lanes of the shipper plus the asset repositioning cost associated with those lanes. Note that stand alone cost of a shipper can be less than the sum of stand alone costs of its lanes due to the synergy between these lanes. Finally “total cost of the collaboration” or equivalently “total budget” is the total cost of satisfying all the demand of a collaboration.
Consider a shippers’ network (Figure 1) with one shipper ($A$) with a lane from node 1 to node 2 and another shipper ($B$) with a lane from node 2 to node 1. Suppose that the cost of covering a lane with a full truckload is equal to 1. We assume the deadhead coefficient, $\theta$, to be 1, then the cost of covering a lane with an empty truck is equal to 1 as well. Then the total cost of covering the lanes in this network is 2. The stand alone cost of each lane in this example is equal to 2. Since each shipper has only one lane in this example stand alone cost of each shipper is equal to 2 as well.

In a proportional cost allocation method, the costs are allocated to the lanes (or to the shippers) proportional to the stand alone costs of the lanes (or shippers). In our example, each shipper has only one lane, therefore both methods yield the same result and allocate a cost of 1 to each shipper.

If a new shipper ($C$) with a lane from node 2 to node 1 enters the collaboration, the total cost of covering the lanes in the network becomes 4. All three lanes (shippers) have the same stand alone cost. Then, the proportional cost allocation method allocates a cost of $\frac{4}{3}$ to each lane (shipper). However, with this allocation, it is not hard to see that shippers $A$ and $B$ (equivalently $A$ and $C$) are better off collaborating on their own with a total cost of 2. Therefore, the proportional cost allocation in this case is not stable and a subgroup, namely $A$ and $B$ (or $A$ and $C$), has an incentive to part company with $C$ (or $B$) and cooperate on its own.

As we consider all possible cost allocations, we conclude that the only allocation where the grand coalition is not threatened by any subgroup of its members is the allocation of $(0, 2, 2)$ to shippers $A$, $B$ and $C$, respectively. Since shipper $A$ has a higher bargaining power compared to the other two shippers, it is expected to be charged less than $B$ and $C$ even though they have the same stand alone costs. Moreover, under any cost allocation method where shipper $A$ has a positive allocation, either one of the other shippers will be willing to cover some of the expenses of shipper $A$ in order to get into a coalition with $A$. However, charging shipper $A$ nothing makes $A$ a free-rider which may not be desirable in a collaboration. Furthermore, in the only stable allocation, where the grand
coalition is maintained together, both shippers B and C are allocated their stand alone costs, so being in a collaboration brings no positive value for these two shippers.

Thus, we conclude that even on a very simple example, basic cost allocation methods such as proportional or stable cost allocation might have some undesirable properties for the shipper collaboration problem and designing a cost allocation method that has all the desired properties may not be possible. Accordingly, one must choose a set of desirable properties and design allocation methods that suit the problem at hand and this is the motivation behind our work.

There are two streams of literature in cooperative game theory and truckload transportation that are relevant to our work. Cooperative game theory studies the class of games in which selfish players form coalitions to obtain greater benefits. A major portion of the literature on cooperative games focuses on finding cost allocations that are stable and allocate the total cost to the members.

We refer the reader to Young (1985) that gives a thorough review of basic cost allocation methods and to Borm et al. (2001) for a survey of cooperative games associated with operations research problems. The games generated by linear programming optimization problems, which are relevant to the game we consider in this paper, are studied in Owen (1975). Owen (1975) proves that for such games a cost allocation that is stable and allocates the total cost can be computed from an optimal solution to the dual of the linear program. Owen (1975) also shows that for the linear production game it considered the converse does not hold and gives an example where such a cost allocation is not included in the optimality set of the dual linear program. These results are further extended in Kalai and Zemel (1982), Samet and Zemel (1984), and Engelbrecht-Wiggans and Granot (1985). Kalai and Zemel (1982) establishes the correspondence of every such cost allocation to an optimal dual solution for a flow game over a simple network. Samet and Zemel (1984) and Engelbrecht-Wiggans and Granot (1985) generalize this result and extend it to some LP games that include the games in which this correspondence is known to exist.

The contribution of this study is to provide a general framework for the cost allocation mechanisms that can be used to manage real-life collaborative truckload transportation networks. There exists a set of papers that join cooperative game theory and classic routing problems, such as Engevall et al. (2004), which consider a vehicle routing game, Sanchez-Soriano et al. (2001), which study transportation games where buyers and sellers are disjoint sets, Granot et al. (1999), which analyze delivery games that are associated with the Chinese postman problem, Derks and Kuipers (1997), which study routing games, and Tamir (1991), which considers continuous and discrete network synthesis games. These papers in general study the existence of cost allocations with well-studied properties from cooperative game theory and propose computational procedures for
finding such allocations. However, to the best of our knowledge there is no literature on cost allocation mechanisms for collaborative logistics problems that identify what the relevant cost allocation properties for the given application should be and develop mechanisms based on these properties. In this paper while searching for a cost allocation method, we do not restrict ourselves only to the concepts from cooperative game theory but also consider the requirements relevant for collaborative truckload transportation networks.

On the other hand, the problem of finding efficient routes, continuous paths and tours, that minimize asset repositioning costs in a collaborative truckload transportation network has been studied by Moore et al. (1991), Ergun et al. (2007) and Ergun et al. (2005). These papers study the underlying optimization problems with side constraints such as temporal and driver restrictions and propose heuristic algorithms for solving them.

The rest of this paper is organized as follows. In Section 1, we briefly discuss well studied cost allocation properties from cooperative game theory. In Section 2, we first give a formal statement of the shippers’ collaboration problem and then develop a stable cost allocation method that allocates the total cost (i.e. an allocation in the core) for the shippers’ collaboration problem using linear programming duality and discuss three other allocation schemes: the nucleolus, the Shapley Value and cross monotonic cost allocations. In Section 3, we develop cost allocations, by relaxing the budget balance and stability properties, that do not allow the allocated costs to be less than the original lane costs and guarantee a reduction from the stand alone cost to each shipper. In Section 4, we investigate how different carrier pricing schemes affect our solution methodologies. In Section 5, we computationally demonstrate how our methods perform on several classes of randomly generated and real-life instances. Concluding remarks are provided in Section 6.

1. Some Cost Allocation Properties

In this section, we discuss some of the well-known cost allocation properties from the cooperative game theory literature.

In a budget balanced, or efficient, cost allocation, the total cost allocated to the members of the collaboration is equal to the total cost incurred by the collaboration. That is, a budget deficit or a surplus is not created. Most allocation methods studied in the literature attempt to find budget balanced allocations. However, in some games, budget balance property conflicts with a more desirable property. In such games, it is possible to seek approximate budget balanced cost allocations that recover at least $\alpha$-percent of the cost (See Pal and Tardos (2003) and Jain and Vazirani (2001) for two applications).
In a stable cost allocation, no coalition of members can find a better way of collaborating on their own. Hence the grand coalition is perceived as stable and is not threatened by its sub-coalitions. Thus, stability is the key concept that holds a collaboration together. The set of cost allocations that are budget balanced and stable is called the core of a collaborative game. For a collaborative game, the core may be empty, that is, a budget balanced and stable cost allocation may not exist. Although stability is a key aspect in establishing a sustainable collaboration, it is not altogether meaningless to consider cost allocation methods with relaxed stability constraints. Due to the costs associated with managing collaborations, limited rationality of the players and membership fees, a sub-coalition might not be formed even though it offers additional benefits to its members. Also, for a collaborative game with an empty core, either budget balance or stability condition should be relaxed in order to find a cost allocation. Therefore, relaxing the stability restriction in a limited way might be acceptable for a cost allocation method.

When there are multiple cost allocations in the core, some of these allocations might be more desirable than the others. One such allocation that is well-studied in the cooperative game theory literature is the nucleolus. Nucleolus, introduced by Schmeidler (1969), is the cost allocation that lexicographically maximizes the minimal gain, the difference between the stand alone cost of a subset and the total allocated cost to that subset, over all the subsets of the collaboration. If the core is nonempty, the nucleolus is included in the core. Nucleolus may still exist even though the core is empty if there exists a cost allocation that is budget balanced and allocates costs to the players that are less than or equal to their stand alone costs.

A well-known cost allocation method is the Shapley Value, which is defined for each player as the weighted average of the player’s marginal contribution to each subset of the collaboration (Shapley 1953). Shapley Value can be interpreted as the average marginal contribution each member would make to the grand coalition if it were to form one member at a time (Young (1985)). Even if the core is non-empty, Shapley Value may not be included in the core.

After the collaboration is initiated, over time there may be potential new members willing to enter the collaboration. Although, it is expected that expansion will be beneficial for the collaboration as a whole due to increased synergies between the members, these benefits may not be distributed evenly among the members. In fact, some of the existing members of the collaboration may be worse off due to the addition of new members under some cost allocation methods. Hence, adding newcomers to the collaboration can break down the collaboration. To avoid this, it is preferable to have a cross monotonic cost allocation method in which any member’s benefit does not decrease
with the addition of a newcomer. A cross monotonic cost allocation is also stable if the allocation is budget balanced (Moulin 1995).

Having a cross monotonic cost allocation is favorable for contractual agreements. If a cross monotonic cost allocation method is used for distributing the costs, then at the contracting stage, a specific cost can be allocated to each member of the collaboration and the members will be assured of never paying more than that value as long as no one leaves the collaboration. See Tazari (2005) for a survey of cross monotonic cost allocation schemes. Unfortunately, as will be discussed, the cross monotonicity property turns out to be a very restrictive one in our setting.

A collaborative game where the players compensate each others’ costs with side payments, is called a transferable payoffs game.

The equal treatment of equals property ensures that two participants of a collaboration have the same allocation if they are identical in every aspect relevant to the allocation problem in consideration. In the shippers’ collaboration problem, equal treatment of equals does not imply that allocated costs of lanes that have the same stand alone cost should be the same. It ensures that the allocated cost to two different shippers is the same if they have lanes with same origin and destination. Throughout the paper, this property is kept as an invariant.

A player $i$ is a dummy player if the incremental cost of adding $i$ to any coalition is zero and the dummy axiom states that a dummy player should get a cost allocation equal to zero. A cost allocation method is additive, if for any joint cost functions, allocation of their sum is equal to the sum of their individual allocations.

Finally, there are some additional cost allocation properties we desire in our setting. As in the example given in the Introduction, under some cost allocation methods it is possible to have members, which pay less than their original lane costs or pay their stand alone cost. We believe that both of these situations might be considered undesirable. We study allocations with “minimum liability” restriction where the shippers are responsible for at least their original lane cost. We also study allocation methods where each shipper is guaranteed an allocation less than its stand alone cost so that being a member of the collaboration offers a “positive benefit” compensating the difficulties (such as integration) of collaborating. Note that, when either of these two restrictions is imposed, it is not possible to have a budget balanced and stable cost allocation for the shippers’ collaboration problem as demonstrated in the example of the Introduction section. Hence, we relax the budget balance and stability properties in a limited way and develop allocations with the above two restrictions.
2. Cost Allocation Methods with Well-known Properties

In this section, we define the "Lane Covering Problem" that seeks to identify the optimal set of cycles covering the lanes in the collaboration and the cooperative game associated with this optimization problem. We also list our assumptions in this section. We then determine a cost allocation mechanism in the core of the shippers’ collaboration problem using linear programming duality. Next, we briefly mention the nucleolus and an alternative cost allocation scheme, the Shapley Value. Finally, we discuss the relationship between the core of the game and cross monotonic cost allocations and devise allocations with this property.

2.1. Problem definition and assumptions

In the shippers’ collaboration problem, we consider a collection of shippers each with a set of lanes that needs to be served. Given the cost of covering each lane and the anticipated repositioning charges, the collaboration’s goal is to minimize the total cost of transportation such that the demand of each shipper in the collaboration is satisfied. The asset repositioning costs depend on the complete lane set of the shippers. As complementarity (synergy) of lanes from different shippers increases (i.e. they form continuous tours with no or minimal deadheading) gains from collaboration and the incentive to collaborate increase.

Next, we summarize our assumptions. First, we assume that every shipper is accepted to be a member of the collaboration even if a shipper does not create a positive value for any other shipper in the collaboration. The question of which shippers should be in the collaboration is beyond the scope of this work. Second, each lane corresponds to a truckload delivery between origin and destination of the lane. We assume that cost of collaborating among the members of the collaboration is negligible. Also, a shipper is not required to submit all of its lanes to the collaboration. Hence, a shipper may exclude a subset of its lanes from the collaboration, and individually negotiate prices with the carrier for those lanes or form another collaboration with other shippers, if it is profitable to do so. We also assume that the lane set consists of only repeatable lanes which means that each shipper has the same lanes to be traversed each period. Finally, we assume that there are no side constraints (like time windows, driver restrictions ... etc.) on the problem.

We will refer to this total transportation costs minimization problem as the Lane Covering Problem (LCP). The LCP is defined on a complete directed Euclidian graph $G = (N, A)$ where $N$ is the set of nodes $\{1, \ldots, n\}$, $A$ is the set of arcs, and $L \subseteq A$ is the lane set requiring service. Let $c_{ij}$ be the cost of covering lane $(i, j)$ with a full truckload. The deadhead coefficient is denoted by $\theta$, where $0 < \theta \leq 1$ hence the asset repositioning cost along an arc $(i, j) \in A$ is equal to $\theta c_{ij}$. Then
the LCP, the problem of finding the minimum cost set of cycles traversing the arcs in $L$, can be solved by finding a solution to the following integer linear program:

$$P: \quad z_L(r) = \min \sum_{(i,j) \in L} c_{ij} x_{ij} + \theta \sum_{(i,j) \in A} c_{ij} z_{ij}$$

subject to

$$\sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} + \sum_{j \in N} z_{ij} - \sum_{j \in N} z_{ji} = 0 \quad \forall i \in N$$

$$x_{ij} \geq r_{ij} \quad \forall (i,j) \in L$$

$$z_{ij} \geq 0 \quad \forall (i,j) \in A$$

$$x_{ij}, z_{ij} \in \mathbb{Z}.$$ 

In $P$, $x_{ij}$ represents the number of times lane $(i,j) \in L$ is covered with a full truckload and $z_{ij}$ represents the number of times arc $(i,j) \in A$ is traversed as a deadhead arc. We let $r_{ij}$ be the number of times lane $(i,j) \in L$ is required to be covered with a truckload and $r$ be the matrix of $r_{ij}$’s. Constraints (2) are flow balance constraints for all the nodes in the network. Constraints (3) ensure that the transportation requirement of each lane in $L$ is satisfied.

Since $P$ is a minimization problem, it is clear that variables $x_{ij}$ will be equal to $r_{ij}$ in any optimal solution and they can be deleted from $P$ resulting in a simpler formulation. However, for the clarity of the following discussions, we work with the above formulation.

In this paper, we focus on the cost allocation game generated by the LCP. In the cooperative game we consider, we designate the lanes in $L$ as the players and try to allocate the total cost of the collaboration among all the lanes served. Since we assume that a shipper may decide to take out a subset of its lanes from the collaboration, allocating the cost to the lanes rather than to the shippers is more appropriate in our setting. The total allocated cost to a shipper is then the sum of the allocated costs to the lanes that belong to the shipper. The characteristic function $z^*_S(r^S)$ (or $z^*_S$) is the optimal cost of covering all the lanes in $S$.

For a given set of lanes, it can be very hard to identify the dynamics of an instance and contribution of each lane to the network without solving an LCP, since there are many subsets of lanes and determining the relationship between these subsets is a formidable challenge. Even with a small modification in the network, optimal set of cycles can change entirely requiring us to solve a new LCP to determine the new optimal set. Therefore, the cost allocations may change significantly with a small change in the input.
2.2. The core of the shippers’ collaboration problem

As stated before, Owen (1975) proves that the optimal dual solutions lead to core cost allocations for LP-games, cooperative games arising from a linear program. Kalai and Zemel (1982) shows that the same result holds for flow games, which can be transformed into the game discussed by Owen (1975). In the view of these results we first prove that LCP is a linear optimization problem, hence showing that the optimal dual solutions of LCP yield cost allocations in the core for the shipper collaboration problem. Next, we show that the converse of the statement also holds; every cost allocation in the core corresponds to an optimal dual solution unlike the game considered in Owen (1975).

**Lemma 1.** The core of the shippers’ collaboration problem with transferable payoffs is non-empty. A cost allocation in the core can be constructed in polynomial time by solving $D$, the dual of the linear programming relaxation of $P$.

**Proof.** $P$ is a minimum cost circulation problem, hence solving its linear relaxation is sufficient to find an integer solution (see Ahuja et al. (1993)). The structure of $P$ is equivalent to the flow problem discussed by Kalai and Zemel (1982), since there exist an arbitrary cost vector $(c)$, service requirement constraints corresponding to each player in the collaboration (constraints (3)), and flow balance constraints for each node in the network (constraints (2)). Therefore the game we consider is an LP game. Hence, the core for this game is nonempty and a cost allocation in the core can be obtained from an optimal dual solution (Owen (1975)).

Let $I_{ij}$ be the dual variables associated with constraints (3) and $y_i$ be the dual variables associated with constraints (2), then the dual of the LP relaxation of $P$ is as follows:

\[
D: \quad d_L(r) = \max \sum_{(i,j) \in L} r_{ij} I_{ij} \quad (6)
\]

\[
s.t. \quad I_{ij} + y_i - y_j = c_{ij} \quad \forall (i,j) \in L \quad (7)
\]

\[
y_i - y_j \leq \theta c_{ij} \quad \forall (i,j) \in A \quad (8)
\]

\[
I_{ij} \geq 0 \quad \forall (i,j) \in L. \quad (9)
\]

Let $\alpha_{ij}$ be the allocated cost for covering lane $(i,j) \in L$ and $(I^*, y^*)$ be an optimal solution of $D$. Then letting $\alpha_{ij} = I_{ij}^* \forall (i,j) \in L$ gives a cost allocation in the core and we can compute $I_{ij}^*$ values by solving $D$ in polynomial time. □

Next we show that every budget balanced and stable cost allocation corresponds to an optimal dual solution. First, a necessary and sufficient condition for a given cost allocation to be stable is given in the next lemma.
Lemma 2. Let \( \alpha \) be a cost allocation (not necessarily budget balanced), then \( \alpha \) is a stable cost allocation if and only if there exists a vector \( y \) that satisfies the linear inequalities

\[
\alpha_{ij} + y_i - y_j \leq c_{ij} \quad \forall (i,j) \in L \quad \text{and} \quad y_i - y_j \leq \theta c_{ij} \quad \forall (i,j) \in A.
\]

**Proof** Let \( C^S \) be the set of cycles that cover the lanes in \( S \subseteq L \) with minimum total cost \( z^*_S \). Hence, \( z^*_S = \sum_{C \in C^S} z(C) \) where \( z(C) \) represents the cost of the cycle \( C \). Also, note that for any cycle \( C \), \( \sum_{(i,j) \in C} (y_i - y_j) = 0 \). For any \( \alpha \) that satisfies the inequalities above, the total cost allocated to any cycle \( C \in C^S \) is less than or equal to the cost of the cycle, since

\[
\sum_{(i,j) \in C \cap S} (\alpha_{ij} + y_i - y_j) + \sum_{(i,j) \in C \cap A \setminus S} (y_i - y_j) \leq \sum_{(i,j) \in C \cap S} c_{ij} + \theta \sum_{(i,j) \in C \cap A \setminus S} c_{ij}
\]

implying

\[
\sum_{(i,j) \in C \cap S} \alpha_{ij} \leq z(C).
\]

Hence over all \( C^S \),

\[
\sum_{C \in C^S} \sum_{(i,j) \in C \cap S} \alpha_{ij} \leq z^*_S.
\]

The inequality above holds for all \( S \subseteq L \), implying that \( \alpha \) is a stable cost allocation.

We prove the necessity of the condition by contradiction. Suppose that for a stable cost allocation \( \alpha \), a vector \( y \) that satisfies the given linear inequalities does not exist. Then the linear program given below is infeasible.

\[
\dot{P} : \max 0
\]

\[
s.t. \quad y_i - y_j \leq c_{ij} - \alpha_{ij} \quad \forall (i,j) \in L
\]
\[
\quad y_i - y_j \leq \theta c_{ij} \quad \forall (i,j) \in A
\]

Therefore, the corresponding dual LP below is either infeasible or unbounded.

\[
\dot{D} : \min \sum_{(i,j) \in L} c_{ij} x_{ij} + \theta \sum_{(i,j) \in A} c_{ij} z_{ij} - \sum_{(i,j) \in L} \alpha_{ij} x_{ij}
\]

\[
s.t. \quad \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} + \sum_{j \in N} z_{ij} - \sum_{j \in N} z_{ji} = 0 \quad \forall i \in N
\]
\[
\quad x_{ij} \geq 0 \quad \forall (i,j) \in L
\]
\[
\quad z_{ij} \geq 0 \quad \forall (i,j) \in A
\]
It is easy to see that \( \bar{D} \) is feasible (\( x = z = 0 \) is a feasible solution). If \( \bar{D} \) is unbounded then its optimal objective function value should be equal to \( -\infty \). Note that due to the flow balance constraints, any feasible solution to \( \bar{D} \) can be decomposed into cycles with positive flows (where this flow is represented by a combination of positive \( x \) and \( z \) values). In order to achieve an objective function value of \( -\infty \), we need to increase the flow over at least one cycle to infinity. Let \( C \) be such a cycle and \( L_C = C \cap L \) and \( A_C = C \cap A \). Then if we increase the flow over cycle \( C \) by one, the objective function value is increased by \( \sum_{(i,j) \in L_C} c_{ij} + \theta \sum_{(i,j) \in A_C} c_{ij} - \sum_{(i,j) \in L_C} \alpha_{ij} \), which is equal to the cost of the cycle minus the allocated cost of the cycle with a stable cost allocation. By definition of stability, this term is always non-negative for any cycle. Hence the optimal objective function for \( \bar{D} \) must be non-negative. Therefore, \( \bar{D} \) is neither unbounded nor infeasible implying that \( \bar{P} \) is feasible. We conclude that for any stable cost allocation there exists a vector \( y \) that satisfies the linear inequalities given above. \( \square \)

Next, we consider the question whether the set of cost allocations in the core for the shippers’ collaboration problem can be completely characterized by the set of optimal solutions to \( D \). That is, whether from each cost allocation in the core, an optimal solution to \( D \) can be constructed and vice versa. If this is the case, then additional properties of these cost allocations can be established by studying the set of optimal solutions to \( D \). In related work, Kalai and Zemel (1982) prove the coincidence of the core with the set of dual optimal solutions for simple network games, Samet and Zemel (1984) extend this result to simple zero-one LP systems, and Engelbrecht-Wiggans and Granot (1985) show the same result for a linear production game with a specific structure. None of these settings includes the shipper collaboration game, since we allow multiple players on the arcs. We extend the above results to the flow game described by the LCP with the following corollary that establishes the equivalence of the core and the optimality set of \( D \).

**Corollary 1.** The core of the transferable payoffs shippers’ collaboration problem is completely characterized by the set of optimal solutions for \( D \).

**Proof** Given Lemma 1, we will only show that every cost allocation in the core corresponds to a feasible solution for \( D \) with the optimal objective function value.

Assume that \( \alpha \) is a cost allocation in the core. Due to the budget balance property of \( \alpha \) and strong duality, \( \sum_{(i,j) \in L} r_{ij} \alpha_{ij} \) is equal to the optimal objective function value for \( D \). \( \alpha \) satisfies constraints (9) since cost allocations cannot be negative.

Due to Lemma 2, for any stable cost allocation, there exists a vector \( y \) that satisfies the linear
inequalities $\alpha_{ij} + y_i - y_j \leq c_{ij} \ \forall (i,j) \in L$ and $y_i - y_j \leq \theta c_{ij} \ \forall (i,j) \in A$. Therefore, for some non-negative matrix $\rho$, we have:

$$\alpha_{ij} + \rho_{ij} + y_i - y_j = c_{ij} \ \forall (i,j) \in L \ \text{and} \ y_i - y_j \leq \theta c_{ij} \ \forall (i,j) \in A.$$ 

Therefore, $(\bar{\alpha}, y)$ where $\bar{\alpha} = \alpha + \rho$ and $\rho \geq 0$ is a feasible solution to $D$. If any $\rho_{ij}$ is positive, then the optimal objective function value of $D$ would be greater than $\sum_{(i,j) \in L} r_{ij} \alpha_{ij}$ which contradicts with the budget balance property of $\alpha$. Therefore, we conclude that $\rho = 0$ and $(\alpha, y)$ is an optimal solution for $D$. $\square$

Next, we discuss the properties of the cost allocations in the core.

If, $r_{ij}$, the number of times that a lane $(i,j)$ must be covered is increased, then cost allocation of that lane, $\alpha_{ij}$, is expected to increase due to dual sensitivity. Equivalently, the cost allocation of lane $(i,j)$ increases as the marginal benefit of lane $(i,j)$ to the network decreases.

If a lane arc $(i,j)$ is covered more times than it is required (i.e. $z_{ij} > 0$), then its allocated cost is less than the original cost of the lane (i.e. $\alpha_{ij} < c_{ij}$), since by complementary slackness, if $z_{ij} > 0$ then $y_i - y_j = \theta c_{ij}$, and hence $\alpha_{ij} = I_{ij} = (1 - \theta)c_{ij}$. Intuitively, if a lane arc is traversed as a deadhead as well, then the contribution of this lane to the collaboration is positive, consequently this lane is charged less than the cost of covering it with a full truckload.

Clearly, any cost allocation constructed from a solution to $D$ satisfies equal treatment of equals property since for each lane $(i,j)$ there is a unique corresponding variable $I_{ij}$. Moreover, the dummy axiom is satisfied as well. Consider a lane $(i,j)$ with zero marginal contribution to each subset of the collaboration. Adding this lane to the collaborative network (i.e. increasing the respective $r_{ij}$ by one) will not increase the total cost of the collaboration. Then, due to LP duality, the dual optimal objective function value does not change even if a coefficient of a dual variable is increased, which means that the corresponding dual variable, $I_{ij}$, and the cost allocation must be zero for that lane.

Let the optimal cycles be the set of simple cycles obtained by decomposing the optimal solution of the LCP. Note that, the total cost of covering the lanes in $L$ is equal to the sum of the costs of the optimal cycles. Given any optimal cycle, total allocated cost of the lanes in that cycle is equal to the cost of covering that cycle, since allocating a cost greater than the cost of covering an optimal cycle would make the allocation unstable and allocating a smaller cost without increasing the cost of another optimal cycle would not be able to recover the entire budget.
2.3. Cost allocations in the core

Recall that the nucleolus is the cost allocation that lexicographically maximizes the minimal gain. Intuitively, the goal of finding the nucleolus of a cooperative game with a non-empty core is to find a cost allocation method in the core that avoids favoring any of the players or subsets in the collaboration as much as possible. Clearly, if there is a unique allocation in the core, then that allocation is the nucleolus of the game. Unfortunately, determining the gain for all the subsets for a collaboration explicitly will take exponential time, hence any nucleolus type cost allocation is not efficiently computable. Therefore, we use a similar approach that is appropriate in our context by devising a cost allocation where the allocated costs are proportional to the original lane costs as much as possible.

Let the “minimum range cost allocation” be the cost allocation in the core which minimizes the deviation of allocated cost of a lane from its original lane cost over all the lanes in the collaboration when there are alternative optimal solutions in the core of the shippers’ collaboration problem. A minimum range core cost allocation can be constructed from a solution to the following LP:

\[
N: \quad N(r) = \min k_2 - k_1
\]
\[\text{s.t.} \quad I_{ij} + y_i - y_j = c_{ij} \quad \forall (i,j) \in L \quad (10)\]
\[y_i - y_j \leq \theta c_{ij} \quad \forall (i,j) \in A \quad (11)\]
\[I_{ij} \geq 0 \quad \forall (i,j) \in L \quad (12)\]
\[\sum_{(i,j) \in L} r_{ij} I_{ij} = d^*_L(r) \quad (13)\]
\[k_2 c_{ij} \geq I_{ij} \geq k_1 c_{ij} \quad \forall (i,j) \in L. \quad (14)\]

A solution is feasible to \(N\) if and only if it is an optimal solution for the linear program \(D\). Hence, a cost allocation constructed from an optimal solution to \(N\) finds an allocation in the core that minimizes the percentage deviation between the allocated cost and the original lane cost over all the lanes in the network. Furthermore, the value of \(k_2\) is bounded between 1 and \(1 + \theta\) and the value of \(k_1\) is bounded between \(1 - \theta\) and 1. Therefore the optimal objective function value, which gives the “range” of the deviation of the cost allocations from the lane costs, is at most \(2\theta\).

This procedure terminates achieving a solution that minimizes the maximum deviation from the original lane costs over all the lanes. Note that this solution may not be a unique optimal solution. Also, different parts of the network may have different characteristics and so this one step algorithm will fail to minimize the maximum deviation value for each part individually. To overcome
these problems, an iterative procedure can be devised that uses the LP above as a subroutine and sequentially allocates costs to lanes in different parts of the network. The idea behind this iterative procedure should be to minimize the range for each lane lexicographically to get a unique cost allocation.

2.4. Shapley Value

The Shapley Value of a lane \((i,j)\) in the shippers’ collaboration game is equal to:

\[
\alpha_{ij} = \sum_{S \subseteq L \setminus \{i,j\}} \frac{|S|! |L \setminus (S \cup \{i,j\})|!}{|L|!} c^{(i,j)}(S)
\]

where \(c^{(i,j)}(S)\) is the marginal cost of adding lane \((i,j)\) to the subset \(S\). To analyze how the Shapley Value behaves in the shippers’ collaboration problem, we consider the simple example from the Introduction section. If only two complimentary lanes are present (i.e. the lanes of shippers \(A\) and \(B\)), then their Shapley Value is equal to 1. If shipper \(C\) enters the network, the total cost of covering the lanes in the network becomes \(3 + \theta\) and the Shapley Values become \(1 - \frac{\theta}{3}\) for shipper \(A\) and \(1 + \frac{2\theta}{3}\) for shippers \(B\) and \(C\).

Shapley Value is the unique allocation method to satisfy three axioms: dummy, additivity and equal treatment of equals. Although Shapley Value may return cost allocations in the core for some instances of the shippers’ collaboration problem, there are many instances where allocations based on Shapley Value are not stable. Furthermore, there exists examples where the total allocated cost for a subset of the lanes is \((1 + \frac{\theta}{2})\) times the subset’s optimal stand alone cost. The allocation in the example is also not stable since \(\alpha_A + \alpha_B = 2 + \frac{\theta}{3} > 2\), so Shapley Value is not in the core.

In general, explicitly calculating the Shapley Value requires an exponential effort and hence it is an impractical cost allocation method unless an implicit technique given the particular structure of the game can be found. For the shippers’ collaboration problem, the Shapley Value and the core both provide budget balanced but not cross monotonic allocations. However, the core allocations are stable. Given that we also do not know of an efficient technique for calculating the Shapley Value for the shippers’ collaboration game, we do not consider it any further in this paper.

2.5. An Important Concept: Cross Monotonicity

In practice, collaborations have a dynamic nature, there may be some new entrants and some members may leave the collaboration over time. Although it is guaranteed that when a new member joins a shippers’ collaboration network the overall benefit will be non-negative, some of the existing members’ costs may increase if the collaboration is not allocating costs based on a cross monotonic
mechanism. This is an undesirable situation since some of the existing members then may oppose to an expansion that increases overall benefits. Furthermore, it would be desirable if a contractual agreement can guarantee a new member that the costs initially allocated to its lanes will not increase unless someone leaves the collaboration. Note that since the players for this collaborative game are the lanes and not the shippers, a new member may also correspond to a new lane from an existing shipper.

Next, we discuss whether the allocations in the core are cross monotonic. Unfortunately, we show that it is not possible to find a cost allocation in the core that is also cross monotonic.

Consider the simple example in Figure 2(a), where there are two lanes from node 2 to node 1 belonging to shippers $B$ and $C$ and one lane from node 1 to node 2 belonging to shipper $A$. Let the cost of covering all the lanes with a truckload be 1 and the deadhead coefficient be $\theta$. If only shippers $A$ and $B$ are present, then the total cost of covering their lanes would be 2. In that case, an allocation in the core must satisfy the following two constraints:

$$\alpha_A + \alpha_B = 2 \quad \text{and} \quad \alpha_A, \alpha_B \leq 1 + \theta.$$ 

The above constraints imply that in a core cost allocation, the allocated cost to each lane can take on any value in the range $(1 - \theta, 1 + \theta)$ as long as the total budget is recovered. If shipper $C$ enters the network, then an allocation in the core must satisfy the following constraints:

$$\alpha_A + \alpha_B + \alpha_C = 3 + \theta, \quad \alpha_A, \alpha_B, \alpha_C \leq 1 + \theta, \quad \alpha_A + \alpha_B + \alpha_A + \alpha_C \leq 2 \quad \text{and} \quad \alpha_B + \alpha_C \leq 2 + 2\theta.$$ 

The dual of the LCP associated with this instance has a unique solution which corresponds to the unique core allocation $\alpha_A = 1 - \theta$ and $\alpha_B = \alpha_C = 1 + \theta$.

Furthermore, if instead of $C$ another shipper, $D$, with a lane from node 1 to 2 enters the collaboration as in Figure 2(b), then an allocation in the core must satisfy the following constraints:
\[ \alpha_A + \alpha_B + \alpha_D = 3 + \theta, \quad \alpha_A, \alpha_B, \alpha_D \leq 1 + \theta, \quad \alpha_A + \alpha_B, \alpha_B + \alpha_D \leq 2 \quad \text{and} \quad \alpha_A + \alpha_D \leq 2 + 2\theta, \]

resulting in the corresponding unique core allocation \( \alpha_A = \alpha_D = 1 + \theta \) and \( \alpha_B = 1 - \theta \). Now we can conclude that no matter which one of the core allocations for shippers \( A \) and \( B \) were chosen initially, an adversary can present either shipper \( C \) or \( D \) to join the network increasing one of the initial allocations. Hence even for this simple instance of the shippers’ collaboration problem there does not exist a cross monotonic cost allocation in the core.

Although this result is discouraging, it is also expected due to the nature of the shippers’ collaboration problem. A new member joining the collaboration can create many alternative tours (cycles) for the cost minimization problem. While the network as a whole may enjoy these alternatives, there may be some individual shippers disadvantaged with the new situation. Since, a newcomer’s effect on the collaboration entirely depends on the initial network structure, having a cost allocation method that benefits all the existing shippers and is also budget balanced is impossible.

As expected, a cost allocation method cannot satisfy all the desired properties for the shippers’ collaboration problem. Hence, alternative cost allocation methods will be of value when different aspects of a successful collaboration are important.

2.5.1. A budget recovery bound for cross monotonic and stable allocations

Since there does not exist a budget balanced and cross monotonic cost allocation for the shippers’ collaboration problem, we next study cross monotonic and stable allocations that recover a good percentage of the total cost. Note that, allocating costs greater than the total budget of the collaboration is not relevant in this context since in such an allocation the stability condition will be violated as well. First, we show that there exists a worst case bound on the percentage of the total budget that any cross monotonic and stable cost allocation method can recover. Then, we present a simple cost allocation method that recovers at least the same percentage for every instance of the problem.

**Lemma 3.** A cross monotonic and stable cost allocation in which the total allocated cost is always less than or equal to the total cost of the collaboration can guarantee to recover at most \( \frac{1}{1+\theta} \) of the total cost.

**Proof** Consider the example in Figure 2 where only shippers \( A \) and \( B \) are present. Let \( \alpha_{12} \) and \( \alpha_{21} \) be the allocated cost to shippers \( A \) and \( B \) respectively with a cross monotonic and stable cost allocation. Note that the total allocated cost to both shippers cannot be greater than the total cost...
of the collaboration, hence $\alpha_{12} + \alpha_{21} \leq 2$. If the number of times lane $(2, 1)$ must be covered, that is $r_{21}$, is increased, then the total cost of the collaboration goes to $(1 + \theta)r_{21}$ as $r_{21}$ approaches infinity. The fraction of the total budget recovered with a cross monotonic and stable cost allocation is less than or equal to $\frac{\alpha_{12} + r_{21}\alpha_{21}}{(1 + \theta)r_{21}}$. Similarly, if $r_{12}$ is increased, then the total cost of the collaboration goes to $(1 + \theta)r_{12}$ as $r_{12}$ approaches infinity and the budget recovery fraction is less than or equal to $\frac{r_{12}\alpha_{12} + \alpha_{21}}{(1 + \theta)r_{12}}$. In order to guarantee the best fraction of the total budget recovered, we need to maximize the minimum fraction $\max\left\{\min\left\{\frac{\alpha_{12} + r_{21}\alpha_{21}}{(1 + \theta)r_{21}}, \frac{r_{12}\alpha_{12} + \alpha_{21}}{(1 + \theta)r_{12}}\right\}\right\}$, therefore both lanes should have a cost allocation equal to 1, ($\alpha_{12} = \alpha_{21} = 1$). Hence, any cross monotonic and stable cost allocation method cannot recover more than $\frac{1}{1 + \theta}$ fraction of the total budget when either $r_{21}$ or $r_{12}$ approaches infinity.

This example represents a worst case instance since there exists a cross monotonic and stable cost allocation that recovers at least the same fraction of the total budget for every instance of the problem. The simple stable cost allocation method where every shipper pays its own lane cost and no one pays for the deadhead costs of the network (i.e. $\alpha_{ij} = c_{ij}$) is cross monotonic, since allocated cost of a lane is independent of other lanes. The amount of recovered budget is equal to $\sum_{(i,j) \in L} c_{ij}$ with this simple allocation mechanism. The total cost of covering all the lanes in the network is composed of two parts: total cost of the lanes and the total deadhead cost. It is easy to see that the total deadhead cost is less than or equal to $\theta$ times the total cost of the lanes, $\theta \sum_{(i,j) \in L} c_{ij}$. Then, this simple stable cost allocation method recovers at least $\frac{1}{1 + \theta}$ fraction of the total budget for every instance. Therefore, we conclude that a cross monotonic and stable cost allocation can guarantee to recover at most $\frac{1}{1 + \theta}$ of the total budget of the collaboration. □

Immorlica et al. (2005) study various problems such as edge cover, vertex cover, set cover, and metric facility location and presents the limitations of cross monotonic cost allocations in budget recovery hence the result above is in line with the results in the literature.

### 2.5.2. A cross monotonic and stable cost allocation method

In the above subsection, we developed an upper bound on the guaranteed fraction of the budget recovered by any cross monotonic and stable cost allocation. We also pointed out a simple cost allocation method which guarantees this upper bound for all instances. However, this simple allocation is ineffective for recovering the best possible fraction of the budget in most instances. Therefore, we attempt to find a cross monotonic and stable cost allocation method that guarantees the same bound and may perform better in some instances.

Recall the minimum range core allocation developed in Subsection 2.3 which minimizes the deviation between the allocated cost and the original lane cost over all the lanes in the network. By
using a similar idea, we develop a procedure that finds cross monotonic and stable cost allocations while maximizing the amount of budget recovered. Cross monotonicity of the allocation is ensured by allocating a cost that is a certain percentage of the stand alone cost to each shipper when the shipper joins the collaboration and never increasing this percentage. In this procedure, $CM$, a modified version of the linear program $D$ is solved every time a new lane enters the collaboration and an optimal solution $(I^*, y^*, k^*)$ is obtained. Then $\alpha_{ij}$, the allocation for lane $(i, j) \in L$ is set equal to $I^*_{ij}$ for all $(i, j)$, recovering $d_L^{CM^*}(r)$ of the total cost.

$$CM : d_L^{CM^*}(r) = \max \sum_{(i,j) \in L} r_{ij}I_{ij}$$

s.t. $I_{ij} + y_i - y_j = c_{ij} \forall (i, j) \in L$ \hfill (18)

$y_i - y_j \leq \theta c_{ij} \forall (i, j) \in A$ \hfill (19)

$I_{ij} = kc_{ij} \forall (i, j) \in L.$ \hfill (20)

Corollary 2. The procedure described above provides a cross monotonic and stable cost allocation method which guarantees to recover at least $\frac{1}{1+\theta}$ of the total budget.

PROOF Due to constraints (18)-(19), a cost allocation $\alpha_{ij} = I^*_{ij}$ where $I^*_{ij}$ is an optimal solution to $CM$ is stable. Also, due to constraints (20), the objective function is equal to $d_L^{CM^*}(r) = \max \sum_{(i,j) \in L} r_{ij}c_{ij}$ because $\sum_{(i,j) \in L} r_{ij}c_{ij}$ is a constant. It is easy to see that $I_{ij} = c_{ij} \forall (i, j) \in L$, $y_i = 0 \forall i \in N$, and $k = 1$ is a feasible solution to $CM$. Therefore, a feasible solution to $CM$ provides a cost allocation that covers at least $\sum_{(i,j) \in L} r_{ij}c_{ij}$. Recall that this value guarantees $\frac{1}{1+\theta}$ fraction of the total budget is recovered.

Next we show that the above procedure provides a cross monotonic cost allocation. Consider adding a new shipper with lane $(i, j)$ to the collaboration. If $(i, j)$ is already in $L$, the constraint set of $CM$ remains the same and the optimal $k$ value and the allocated costs to lanes do not change. If $(i, j)$ is not in $L$, then additional constraints (stability conditions and the constraint $I_{ij} = kc_{ij}$) are introduced which may further restrict the $k$ value. Since $CM$’s feasible region becomes smaller, the value of $k$ (and so the allocated cost of the lanes) does not increase with additional members in the collaboration. □

This method in general is expected to recover a greater fraction of the total budget than allocating only the lane costs, however the budget recovery is still bounded with $\frac{1}{1+\theta}$ in the worst case. We believe that allocation methods with such recovery bounds are impractical and will not be used in...
the industry. However, due to the nature of the problem and the cross monotonicity property, no cost allocation method can do any better.

Another drawback of this method is observed when a cycle consists of only the lane arcs. Recall that the total cost allocated to a cycle should be less than or equal to the cost of the cycle due to the stability condition. In that case, the procedure above yields $k^* = 1$, which means that every lane in the collaboration is allocated only the lane costs and no one pays for the deadhead costs. In a large collaboration with a considerable number of lanes, this phenomena is most likely to happen as we observed in our computational study. Similar to the minimum range core allocation developed in Subsection 2.3, an iterative procedure that sequentially allocates costs to the lanes in different parts of the network may be employed to eliminate this drawback.

3. Cost Allocation Methods with Additional Desirable Properties

In the previous sections, we have considered the standard cost allocation properties from cooperative game theory and proved that there does not exist a cost allocation method that is in the core and cross monotonic. However, for the shippers’ collaboration problem there are other desirable cost allocation properties besides these well-known concepts. A shipper joins a collaborative network to reduce its total transportation expenses by reducing asset repositioning costs incurred while covering its lanes with a truckload. Hence a collaborative transportation procurement network should not allocate the entire stand alone cost or a cost that is less than the truckload lane cost to any shipper. We will first consider a “minimum liability” cost allocation where every lane pays at least its original truckload lane cost and then a “positive benefit” cost allocation where every lane is ensured to have a certain discount from its stand alone cost.

In following subsections, we attempt to develop methods that provide cost allocations satisfying these new properties while violating the formerly discussed properties minimally. That is, we first relax either budget balance or stability conditions and impose the restrictions associated with the new properties, and then attempt to find an allocation with a minimum deviation from the relaxed property.

3.1. Cost allocation methods with the minimum liability restriction

In the previous sections, we considered cost allocations with transferable payoffs where the allocated cost of some of the lanes are less than their original lane cost. In this context, not only the asset repositioning costs but also the total lane covering costs are distributed among all the shippers without any restrictions leading to the existence of “free-riders” with zero allocated costs.

In a real world situation, shippers may not be pleased to cover some other shipper’s truckload expense and to have a free-rider in the collaboration. Therefore, we next develop a cost allocation
method where shippers are responsible at least for their truckload lane costs and only the total deadhead cost is distributed among the members of the collaboration. First, we show that when such a restriction is imposed, the core of the shippers’ collaboration game may be empty.

**Lemma 4.** The core of the shippers’ collaboration problem with the minimum liability restriction may be empty.

**Proof.** We provide an instance of the problem with an empty core. Consider the simple example from the Introduction section where shipper A has a lane from node 1 to node 2 and shippers B and C each have lanes from node 2 to node 1. The cost of covering each lane with truckload is equal to 1 and as a deadhead is $\theta$. In a minimum liability cost allocation, $\alpha_{12} \geq 1$ and $\alpha_{21} \geq 1$. If $\alpha_{12} = \alpha_{21} = 1$, then the cost allocation is not budget balanced. If either $\alpha_{12}$ or $\alpha_{21}$ is greater than 1, then the cost allocation is not stable. Hence the core for the above instance of the shippers’ collaboration problem is empty. $\square$

To determine whether the core of a given instance of the shippers’ collaboration problem with the minimum liability restriction is empty, the linear program $D$ may be solved with the additional constraint $I_{ij} \geq c_{ij} \quad \forall (i,j) \in L$. If the objective function value for this constrained version of $D$ is equal to the total cost of covering all the lanes, then we conclude that the core is non-empty.

As the core may be empty for the cost allocations with the minimum liability restriction, we may consider relaxing the budget balance or the stability conditions to design cost allocation methods with the minimum liability restriction. However, any budget balanced relaxed cost allocation method has a lower bound on the fraction of the budget that can be recovered that is far from promising for its use in practice. A minimum liability cost allocation with the best possible budget recovery can be found by solving the following LP:

$$BB: d^BB_L(r) = \max \sum_{(i,j) \in L} r_{ij} I_{ij}$$

subject to:

$$I_{ij} + y_i - y_j \leq c_{ij} \quad \forall (i,j) \in L$$

$$y_i - y_j \leq \theta c_{ij} \quad \forall (i,j) \in A$$

$$I_{ij} \geq c_{ij} \quad \forall (i,j) \in L.$$  

**Lemma 5.** The cost allocation obtained by letting $\alpha_{ij} = I^*_{ij} \quad \forall (i,j) \in L$, where $I^*$ is the optimal solution for the linear program $BB$, provides a stable cost allocation for the shippers’ collaboration problem with the minimum liability restriction that recovers the maximum possible percentage of the total budget. The lower bound on the fraction of total cost recovered is $\frac{1}{1+\theta}$.
PROOF Due to Lemma 2 and constraints (24), any stable minimum liability cost allocation is a feasible solution to $BB$. Hence the optimal solution to $BB$ provides an allocation that maximizes the total budget recovered among all the stable minimum liability cost allocations. Since $\sum_{(i,j) \in L} r_{ij} \alpha_{ij} = \sum_{(i,j) \in L} r_{ij}^* I_{ij}^* \geq \sum_{(i,j) \in L} r_{ij} c_{ij}$, we conclude that the second part of the lemma is satisfied as well. Note that this lower bound is tight for the example discussed in the proof of Lemma 3. □

Cost allocations with a minimum percentage deviation from stability Stability is the key property that holds a collaboration together and without this property a collaboration is under the risk of collapsing. On the other hand, in many real life situations, this threat is not as immediate as we conclude from a theoretical perspective. Even if there exists a sub-coalition that is better off by collaborating on its own, this usually does not mean that the grand coalition will break down. In real life, due to insufficient information sharing, limited rationality of the players, cost of collaborating, and contractual agreements many of the sub-coalitions are not formed even though they create additional benefits for its members. Therefore, relaxing the stability conditions in a limited way can be justified.

Relaxing the stability condition in order to find a budget balanced minimum liability cost allocation is not as straightforward as relaxing the budget balance property. There is an exponential number of subsets to consider while seeking an allocation method which has minimum deviation from stability over all these subsets. For any subset of the collaboration, the deviation from stability can be defined as the percent deviation or the fixed value deviation from stand alone cost of the subset. We next develop a procedure based on modifications to the linear program $D$ that finds cost allocations with minimum deviation from stability.

In the method we propose, the objective is to minimize the maximum percentage deviation of the allocated cost of a subset from its stand alone cost over all the subsets of the collaboration while making sure that the total budget is recovered and the allocated cost of a lane is greater than or equal to its truckload cost. This method is a quick an effective method but most importantly it provides a minimum liability cost allocation with minimum possible percentage deviation.

First, we show that by generalizing Lemma 2, we can calculate the maximum percentage deviation from stability of a given cost allocation efficiently, that is without actually considering exponentially many subsets.

THEOREM 1. Let $\alpha$ be a cost allocation. For a given scalar, $k$, if there exists a vector $y$ that satisfies the linear inequalities
Consider the primal dual LP pair given below for a cost allocation

More specifically, we show that there exists a cycle

implying that the deviation from stability for \( \alpha \) over all subsets is at most \( k \), hence the percentage deviation is \( 100 \times k \). Let \( k^* \) be the minimum of such \( k \) values, then there exists a \( S \subseteq L \) such that \( (1+k^*)z^*_S = \sum_{(i,j) \in S} \alpha_{ij} \).

**PROOF** Proof is similar to the proof of Lemma 2. Let \( C^S \) be the set of cycles that traverse the lanes in \( S \subseteq L \) with minimum total cost, \( z^*_S \). Hence, \( z^*_S = \sum_{C \in C^S} z(C) \) where \( z(C) \) represents the cost of the cycle \( C \). Also, note that for any cycle \( C \), \( \sum_{(i,j) \in C} (y_i - y_j) = 0 \). The total cost allocated to \( S \) with cost allocation \( \alpha \) is less than or equal to \( (1+k) \times z^*_S \) since for any cycle \( C \) in \( C^S \),

\[
\sum_{(i,j) \in C \cap S} (\alpha_{ij} + y_i - y_j) + \sum_{(i,j) \in C \cap A \setminus S} (y_i - y_j) \leq \sum_{(i,j) \in C \cap S} c_{ij}(1+k) + \theta \sum_{(i,j) \in C \cap A \setminus S} c_{ij}(1+k)
\]

implying

\[
\sum_{(i,j) \in C \cap S} \alpha_{ij} \leq z(C) \times (1+k).
\]

Then

\[
\sum_{C \in C^S} \sum_{(i,j) \in C \cap S} \alpha_{ij} \leq z^*_S (1+k),
\]

implying that the deviation from stability for \( \alpha \) is at most \( k \) for \( S \). Since \( S \) is an arbitrary subset of \( L \), we conclude that the deviation from stability for \( \alpha \) over all subsets is at most \( k \).

Next, we prove by contradiction that there exists a \( S \subseteq L \) such that \( (1+k^*)z^*_S = \sum_{(i,j) \in S} \alpha_{ij} \).

More specifically, we show that there exists a cycle \( C \) such that \( (1+k^*)z(C) = \sum_{(i,j) \in C \cap L} \alpha_{ij} \).

Consider the primal dual LP pair given below for a cost allocation \( \alpha \) and scalar \( k \).

\[
\hat{P}(\alpha, k) : \max 0
\]

\[
s.t. \ y_i - y_j \leq c_{ij}(1+k) - \alpha_{ij} \ \forall (i,j) \in L
\]

\[
\quad \ y_i - y_j \leq \theta c_{ij}(1+k) \ \forall (i,j) \in A.
\]

\[
\hat{D}(\alpha, k) : \min (1+k) \sum_{(i,j) \in L} c_{ij}x_{ij} + (1+k)\theta \sum_{(i,j) \in A} c_{ij}z_{ij} - \sum_{(i,j) \in L} \alpha_{ij}x_{ij}
\]

\[
s.t. \ \sum_{j \in N} x_{ij} - \sum_{j \in N} x_{ji} + \sum_{j \in N} z_{ij} - \sum_{j \in N} z_{ji} = 0 \ \forall i \in N
\]

\[
\quad x_{ij} \geq 0 \ \forall (i,j) \in L
\]

\[
\quad z_{ij} \geq 0 \ \forall (i,j) \in A.
\]
Note that for a given allocation $\alpha$ and its corresponding $k^*$, both $\tilde{P}(\alpha, k^*)$ and $\tilde{D}(\alpha, k^*)$ are feasible and bounded due to definition of $k^*$. Now suppose that for $\alpha$ and $k^*$ all the cycles satisfy

$$(1 + k^*) \sum_{(i,j) \in C \cap L} c_{ij} + (1 + k^*)\theta \sum_{(i,j) \in C \cap A} c_{ij} > \sum_{(i,j) \in C \cap L} \alpha_{ij}$$

that is

$$(1 + k^*) z(C) > \sum_{(i,j) \in C \cap L} \alpha_{ij}.$$ 

Therefore, for some $\bar{k} < k^*$, any cycle satisfies

$$(1 + \bar{k}) z(C) \geq \sum_{(i,j) \in C \cap L} \alpha_{ij}.$$ 

Note that $\tilde{D}(\alpha, \bar{k})$ is feasible. Furthermore, due to the above inequality and similar arguments as in the proof of Lemma 2, by increasing flow over any cycle, we cannot decrease the optimal objective function of $\tilde{D}(\alpha, \bar{k})$ below 0. Hence, $\tilde{D}(\alpha, \bar{k})$ is also bounded. Consequently, $\tilde{P}(\alpha, \bar{k})$ is also feasible and bounded contradicting with the minimality of $k^*$. Therefore we conclude that there exists at least one cycle $C$ such that the cost allocated to cycle $C$ is equal to $(1 + k^*)$ times the stand-alone cost of cycle $C$, $(1 + k^*) z(C) = \sum_{(i,j) \in C \cap L} \alpha_{ij}$. □

For a given cost allocation, maximum deviation from stability, $k^*$, can be determined by solving a linear program with the objective of minimizing $k$ over the linear inequalities given above. Next, we continue with the method.

**Method SR:**

**Step 1:** Find the minimum total cost of covering all the lanes in the network by solving either $P$ or $D$. Let this cost be $z^*_L(r)$.

**Step 2:** Using the solution in Step 1, formulate and solve the LP below:

$$SR: \quad d^{SR}_L(r) = \min_k \quad (25)$$

s.t.   $$\sum_{(i,j) \in L} r_{ij} I_{ij} = z^*_L(r) \quad (26)$$

$$I_{ij} \geq c_{ij} \quad \forall (i,j) \in L \quad (27)$$

$$I_{ij} + y_i - y_j \leq (1 + k)c_{ij} \quad \forall (i,j) \in L \quad (28)$$

$$y_i - y_j \leq \theta c_{ij} (1 + k) \quad \forall (i,j) \in A. \quad (29)$$

**Step 3:** Let $(I^*, y^*, k^*)$ be an optimal solution to $SR$. Then let $\alpha_{ij} = I^*_{ij} \forall (i,j) \in L$. 
THEOREM 2. The procedure above gives a budget balanced cost allocation with the minimum liability restriction where the optimal k value, $k^*$, is the maximum deviation from stability over all the subsets of the collaboration. $k^*$ represents the minimum possible deviation from stability over all the budget balanced cost allocations with the minimum liability restriction. The value of $k^*$ is bounded from above by $\frac{\theta}{2}$ which, in fact, is the best upper bound.

PROOF By construction $SR$ provides a budget balanced cost allocation that satisfies the minimum liability restriction. Also due to Theorem 1, the maximum deviation of allocated cost from the stand alone cost over all subsets is exactly equal to $k^*$.

For the second part of the theorem, we note that all the cost allocations that satisfy the minimum liability and budget balanced restrictions are feasible for $SR$. Hence due to Theorem 1, the cost allocation found by Method $SR$ will have the minimum possible deviation from stability over all the budget balanced cost allocations with the minimum liability restriction.

For the last part of the theorem, we first construct a feasible solution to the linear program $SR$ from the optimal solution to $D$ with a k value equal to $\frac{\theta}{2}$. By doing so, we prove that $SR$ is always feasible and $k^*$ is bounded from above by $\frac{\theta}{2}$, since $SR$ is a minimization problem.

Let $(\bar{I}, \bar{y})$ be the optimal solution to $D$, the dual of the lane covering problem. Due to the constraints $I_{ij} = c_{ij} + y_{ij} - y_i \forall (i, j) \in L$, $\bar{I}_{ij} \geq c_{ij}$ if $(\bar{y}_j - \bar{y}_i)$ is nonnegative, otherwise $\bar{I}_{ij} < c_{ij}$. Let’s partition the lane set into two sets: $L_1 = \{(i, j) : \bar{I}_{ij} \geq c_{ij}\}$ and $L_2 = \{(i, j) : \bar{I}_{ij} < c_{ij}\}$.

Now, we construct a feasible solution to $SR$ from $(\bar{I}, \bar{y})$. Let $\hat{I}_{ij}$ and $\hat{y}_i$ be such that:

$$\hat{I}_{ij} = \bar{I}_{ij} - \rho_{ij} \geq c_{ij} \forall (i, j) \in L_1,$$

$$\hat{I}_{ij} = c_{ij} \forall (i, j) \in L_2,$$

$$\hat{y}_i = \frac{\bar{y}_i}{2} \forall i$$

where $\rho_{ij}$’s are selected so that $\sum_{(i,j) \in L} r_{ij} \hat{I}_{ij} = z^*_L(r)$. Note that it is possible to find such $\rho \geq 0$ because $\sum_{(i,j) \in L} r_{ij} c_{ij} \leq z^*_L(r)$. Also, note that $\hat{y}_i - \hat{y}_j = \frac{\bar{y}_i}{2} - \frac{\bar{y}_j}{2} \leq \frac{\theta}{2} c_{ij} \forall (i, j) \in A$ due to constraint (8) of $D$ and $c_{ij} = c_{ji} \forall (i, j) \in L$ (Euclidian graph).

We claim that $(\hat{I}, \hat{y}, \frac{\theta}{2})$ is a feasible solution to $SR$. Constraints (26) and (27) are satisfied due to construction of $(\hat{I}, \hat{y})$. Constraints (29) are satisfied since

$$\hat{y}_i - \hat{y}_j \leq \frac{\theta}{2} c_{ij} < \theta c_{ij}(1 + \frac{\theta}{2}) \forall (i, j) \in A.$$

Constraints (28) are partitioned into two sets with respect to $L_1$ and $L_2$ and are satisfied as shown below since $\hat{y}_i - \hat{y}_j \leq \frac{\theta}{2} c_{ij} \forall (i, j) \in A$, $c_{ij} = c_{ji} \forall (i, j) \in L$ and $(\hat{I}, \hat{y})$ is the optimal solution to $D$. 
\[ \hat{I}_{ij} + \hat{y}_i - \hat{y}_j = \bar{I}_{ij} - \rho_{ij} + \bar{y}_i - \bar{y}_j = c_{ij} + \bar{y}_j - \bar{y}_i - \rho_{ij} + \hat{y}_i - \hat{y}_j \]

\[ = c_{ij} + \bar{y}_j - \bar{y}_i - \rho_{ij} \leq c_{ij}(1 + \frac{\theta}{2}) \quad \forall (i,j) \in L_1 \quad \text{and} \]

\[ \hat{I}_{ij} + \hat{y}_i - \hat{y}_j = c_{ij} + \hat{y}_j - \hat{y}_i - \rho_{ij} \leq c_{ij}(1 + \frac{\theta}{2}) \quad \forall (i,j) \in L_2. \]

Thus, we conclude that \((\hat{I}, \hat{y}, \frac{\theta}{2})\) is a feasible solution and the optimal objective function value of \(SR\) is bounded above by \(\frac{\theta}{2}\).

Moreover, \(\frac{\theta}{2}\) is the best upper bound, that is, there exists an instance where the optimal value of \(k\) is equal to \(\frac{\theta}{2}\). Consider the example in Figure 3 where there is one lane from node 1 to node 2 and an infinite number of lanes from node 2 to node 1. For this instance, the unique budget balanced minimum liability cost allocation allocates a cost of 1 to the lane from node 1 to 2 and \(1 + \theta\) to each lane from node 2 to 1. Given these values, allocated cost to each cycle consisting of arcs (1, 2) and (2, 1) is equal to \(2 + \theta\) which deviates by \(\frac{\theta}{2}\) from the stand alone cost of the corresponding subset of lanes.

\[ \Box \]

3.2. Cost allocation methods with guaranteed positive benefits

A collaboration is established to create benefits for its members, therefore any given shipper expects to have positive gains when entering a shippers’ collaboration. We next discuss cost allocation methods, which ensure that each shipper is charged less than its stand alone cost or equivalently that each lane in the collaborative network has a “positive benefit”. Similar to the cross monotonicity property, such methods are desirable in contracting stages since promising each player at least a certain reduction from its stand alone costs increases the attractiveness of the collaboration.

As we have presented previously there exists instances of the shippers’ collaboration problem where all the cost allocations in the core allocates the stand alone costs to some of the shippers. In such situations, there exists a set of shippers who have no gains from being in the collaboration. This might lead the shippers to leave the collaboration (or at least move out some of their lanes from the collaborative network) to avoid costs and problems associated with coordinating with
other members. It may not be possible to find a budget balanced and stable cost allocation if we impose that the allocation should also provide a positive benefit to each member. Hence, we study positive benefit cost allocation mechanisms where one of these conditions is relaxed.

For instances where there are very limited or no synergies among the lanes of a collaborative network, a cost allocation with positive benefits to its members will not be possible without a huge budget deficit as should be expected. In such situations, the collaborations are not expected to survive.

Offering any shipper an allocated cost less than its stand alone cost may not be enough for the stability of the collaboration, since there may be other possibilities for the shipper to exploit, such as sub-coalitions, offering even higher benefits. However, it is not unreasonable to assume that the members of a collaborative network have limited rationality or equivalently each shipper cannot be aware of the structure of the entire network in order to exploit it. Hence, a collaboration ensuring a certain percentage reduction from the stand alone costs may still be stable.

3.2.1. Relaxing the budget balance property First, we develop a stable cost allocation method with positive benefits while relaxing the budget balance property. Let \((I^*, y^*)\) be an optimal solution to the following linear program \(B\). Then a stable allocation \(\alpha\) can be constructed by letting \(\alpha_{ij} = I^*_{ij}\) \(\forall (i, j) \in L\), where the allocated cost of any lane is at most \(100 \times \sigma\) percent of its stand alone cost and the fraction of the budget recovered is maximized.

\[
\begin{align*}
B : d^B_L(r) &= \max \sum_{(i, j) \in L} r_{ij} I_{ij} \\
\text{s.t.} & \quad I_{ij} + y_i - y_j \leq c_{ij} \quad \forall (i, j) \in L \\
& \quad y_i - y_j \leq \theta c_{ij} \quad \forall (i, j) \in A \\
& \quad I_{ij} \leq (1 + \theta)c_{ij} \sigma \quad \forall (i, j) \in L.
\end{align*}
\]

Note that, here we assume \(0 < \sigma < 1\) is pre-specified by the collaboration and given the specified \(\sigma\) value, \(B\) is solved to get an allocation with at least \(100 \times \sigma\) percent savings from the stand alone cost of each lane.

Next we try to find an upper bound for the budget deficiency of the cost allocation constructed with this method. Let \(\hat{\alpha}\) be a cost allocation in the core for the shippers’ collaboration problem. Let \(L_c\) be the set of lanes such that \(\hat{\alpha}_{ij} > (1 + \theta)c_{ij}\sigma\). If we update the allocation of the lanes in \(L_c\) by decreasing their allocations so that we get a feasible solution for \(B\), then the budget deficiency of this allocation will be at most

\[
\sum_{(i, j) \in L_c} r_{ij} c_{ij}(1 + \theta)(1 - \sigma) \leq \sum_{(i, j) \in L} r_{ij} c_{ij}(1 + \theta)(1 - \sigma) = (1 - \sigma) \sum_{(i, j) \in L} a_{ij}
\]
where $a_{ij} = (1 + \theta)c_{ij}$ is the stand alone cost of lane $(i, j)$. If the pre-specified $\sigma$ values are different for different lanes then the budget deficiency of this allocation will be at most

$$\sum_{(i,j) \in L} r_{ij}c_{ij}(1 + \theta)(1 - \sigma_{ij}) = \sum_{(i,j) \in L} (1 - \sigma_{ij})a_{ij}.$$ 

Note that this bound is tight for a collaborative network with no synergies between its lanes. For the above constructed allocations the remaining fraction of the budget not recovered may be too large to be distributed to the members of the collaboration in other ways. Therefore, we consider relaxing the stability property next.

### 3.2.2. Relaxing the stability property

Now we develop a method, which provides a positive benefits allocation that recovers the entire budget and has minimum percentage instability over all the subsets of the collaboration. We use a similar methodology to Method SR of Subsection 3.1 and first solve the following linear program:

$$SS : d^SS_{SA}(r) = \min k$$

s.t. $I_{ij} + y_i - y_j \leq c_{ij}(1 + k) \forall (i, j) \in L$ \hfill (35)

$y_i - y_j \leq \theta c_{ij}(1 + k) \forall (i, j) \in A$ \hfill (36)

$$\sum_{(i,j) \in L} r_{ij}I_{ij} = z^*_L(r)$$ \hfill (37)

$I_{ij} \leq (1 + \theta)c_{ij}\sigma \forall (i, j) \in L.$ \hfill (38)

As before we assume that $\sigma$ is pre-specified. If we let $\alpha_{ij} = I^*_{ij} \forall (i,j) \in L$ where $(I^*, y^*, k^*)$ is an optimal solution to $SS$, then $\alpha$ is a budget balanced allocation which allocates at most $100 \times \sigma$ percent of its stand alone cost to each lane and the maximum instability over all the subsets of the collaboration is $100 \times k^*$ percent. However in this case $SS$ might be infeasible. In fact, for collaborative networks with little synergy between its lanes the allocation problem might be infeasible even for a small $\sigma$ value. A rough lower bound for $\sigma$ which ensures the feasibility of $SS$ is equal to $z^*_L(r) / \sum_{(i,j) \in L} a_{ij}$. The optimal $k$ depends on the pre-specified $\sigma$ value and the instance considered.

### 4. Cost Allocations Under Different Carrier Cost Structures

Throughout the paper we have assumed that the carrier’s cost structure is such that if $c_{ij}$ is the cost of traversing an arc with a full truckload then the cost of traversing the same arc as a deadhead is equal to $\theta c_{ij}$ for some $0 < \theta \leq 1$. In this section, we present two different cost structures that might be used by the carriers in the industry and discuss how these affect the allocation methods described above.
4.1. Multiple deadhead coefficients

The deadhead costs over the arcs of a carrier’s network may not be homogenous and the associated cost of traversing an arc \((i, j)\) might be equal to \(\theta_{ij}c_{ij}\). Such a cost structure captures the fact that different parts of the network have different characteristics such as load imbalance.

The existence of multiple deadhead coefficients does not affect much of the procedures described above. First of all, the LCP can be solved with the same minimum cost circulation linear program by replacing \(\theta\) with \(\theta_{ij}\) appropriately. When \(D\), the dual of the LP relaxation of the LCP is considered, the only change occurs in the constraints \(y_i - y_j \leq \theta c_{ij} \forall (i, j) \in A\). Updating this constraint set with \(y_i - y_j \leq \theta_{ij}c_{ij} \forall (i, j) \in A\) and solving the dual problem provides a cost allocation in the core. Thus, all the procedures described above remain the same, however, the bounds should be updated accordingly.

4.2. Fixed deadhead costs

A carrier might charge a fixed cost for each deadhead rather than a variable cost based on the truckload cost of the arc. Let \(\theta_f\) be this fixed deadhead charge. In this case, the LCP is modified by replacing the objective function with

\[
\min \sum_{(i,j) \in L} c_{ij} x_{ij} + \theta_f \sum_{(i,j) \in A} z_{ij}.
\]

Accordingly the dual of the LCP is modified by replacing the constraints (8) with \(y_i - y_j \leq \theta_f \forall (i, j) \in A\). Note that the “fixed deadhead cost” problem can be converted to the “multiple deadhead cost” problem by letting \(\theta_{ij} = \frac{\theta_f}{c_{ij}}\). Then the constraints \(y_i - y_j \leq \theta_f \forall (i, j) \in A\) are replaced with \(y_i - y_j \leq \theta_{ij}c_{ij} \forall (i, j) \in A\) and the rest follows. Thus, the procedures remain the same and the bounds should be modified.

5. Computational Study

We have carried out a computational study to evaluate the performance of the methods we have discussed on randomly generated instances and real-life instances. The real-world data is provided by a strategic sourcing consortium for a $14 billion dollar sized US industry. The company name is kept confidential at the company’s request. Although we presented theoretical bounds for the key performance metrics for most of the allocation methods discussed above, we believe performance of these methods will be significantly better on average.

The random instances we use are generated within a 1,000 \(\times\) 1,000 square with different number of nodes, average number of edges per node, number of clusters and ratio of cluster points to total points. The first two parameters control the size of the instance generated. Clusters represent the
dense regions with many supply or demand points such as metropolitan areas and the cluster ratio
controls how many of the total points fall within a cluster. Clusters are uniformly distributed in the
square and each point in a cluster is generated by using a standard Normal distribution. Remaining
points are distributed uniformly across the map. We do not generate any lanes between any two
points of a cluster.

Finally, we generate a set of instances simulating a supply chain structure. In a “supply chain
instance,” all the points generated are divided into three categories: Suppliers, plants and dis-
tribution centers, and customers (with 0.2, 0.1, 0.7 fraction of the points) and lanes are directed
from suppliers to plants and DC’s and from plants and DC’s to customers. Lanes among plants
and DC’s are also allowed. The supply chain instances are more relevant when the collaboration is
intra-firm where as the random instances represent an inter-firm collaborative network.

We generate instances with 50, 100, 200 and 500 points, 2, 5 and 10 average lanes per point,
5 and 10 clusters and 0, 0.3 and 0.6 cluster ratios. For each combination of these parameters,
we generate two random instances, with and without a supply chain structure. We perform each
experiment with the deadhead coefficients 0.25, 0.5, 0.75 and 1. In order to investigate the effect
of each parameter on the key performance indicators, we take the average of the results on the
576 instances generated with respect to these parameters. These results are presented in Table 1
and Table 2. We remind the reader that there exist a drawback of the cross monotonic and stable
cost allocation method (CM). Recall that when a cycle consists of only lane arcs, the procedure
CM yields a cost allocation where every lane in the collaboration is allocated only the lane costs
and this phenomena is most likely to happen in large instances. Thus, the fraction of the budget
recovered with the CM method is not a meaningful measure of performance for large instances
and because of this we do not present computational results on the CM method.

First, all of our procedures are very efficient and complete within a few minutes. For example,
for one of the largest instances tested (500 nodes and 10 average lanes per node) the solution
times for the procedures such as primal (P), dual (D), minimum range cost allocation (N), cross
monotonic cost allocation (CM), budget recovery (BB), and Method SR (SR) are 20.27, 44.64,
45.35, 10.56, 38.63, and 207.23 seconds respectively. The solution times of the procedures with the
positive benefits restriction highly depend on the pre-specified $\sigma$ values. Specifically, when there is
no feasible solutions for an instance for a given $\sigma$ value, the procedures take a long time to prove
infeasibility.

In Table 1, the column “DH %” presents the percentage of the total deadhead cost to the total
cost of the instance. The column “Range” represents the optimal objective function value of the core
cost allocation method discussed in Subsection 2.3, which measures the range of the deviation of the cost allocations from the lane costs. In order to make the average value of this performance metric over different deadhead coefficients meaningful, the range values are normalized by respective $\theta$ values. The values in the next two columns are for minimum liability allocation methods: “MRBR” column represents the percentage of budget recovered when the budget balance property is relaxed and “MRSTMSR” column presents the percent instability value for Method SR. Finally, “INSNR” column represents the percent instability value of the proportional cost allocation scheme, a cost allocation method currently used in practice. The values in this column are obtained by solving a linear program described in Theorem 1.

On the other hand, the first row of Table 1 presents the average values of each performance indicator over all 576 instances. In the next four rows, the average of performance indicators with respect to different deadhead coefficient values are given. Similarly, the average values of the indicators are presented with respect to different number of nodes in rows 6-9, with respect to average number of lanes per node in rows 10-12, with respect to different number of clusters in rows 13-14, with respect to different cluster ratios in rows 15-17 and finally with respect to the supply chain instance indicator in rows 18-19. Finally, the last two rows present the maximum and minimum values of the performance indicators over all the instances generated.

Table 1  **Average values of performance indicators of instances with respect to different parameter settings.**

<table>
<thead>
<tr>
<th></th>
<th>DH %</th>
<th>Range</th>
<th>MRBR</th>
<th>MRSTMSR</th>
<th>INSPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>7.82</td>
<td>1.97</td>
<td>95.66</td>
<td>3.55</td>
<td>8.93</td>
</tr>
<tr>
<td>$\theta$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.25</td>
<td>3.42</td>
<td>1.97</td>
<td>98.12</td>
<td>1.47</td>
<td>3.58</td>
</tr>
<tr>
<td>0.5</td>
<td>6.52</td>
<td>1.97</td>
<td>96.40</td>
<td>2.89</td>
<td>7.15</td>
</tr>
<tr>
<td>0.75</td>
<td>9.37</td>
<td>1.97</td>
<td>94.80</td>
<td>4.26</td>
<td>10.72</td>
</tr>
<tr>
<td>1</td>
<td>11.98</td>
<td>1.97</td>
<td>93.32</td>
<td>5.59</td>
<td>14.29</td>
</tr>
<tr>
<td># of Nodes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>11.68</td>
<td>1.96</td>
<td>93.75</td>
<td>5.11</td>
<td>13.92</td>
</tr>
<tr>
<td>100</td>
<td>8.30</td>
<td>1.95</td>
<td>95.78</td>
<td>3.43</td>
<td>9.43</td>
</tr>
<tr>
<td>200</td>
<td>6.84</td>
<td>1.98</td>
<td>95.90</td>
<td>3.37</td>
<td>7.60</td>
</tr>
<tr>
<td>500</td>
<td>4.47</td>
<td>1.99</td>
<td>97.21</td>
<td>2.30</td>
<td>4.78</td>
</tr>
<tr>
<td>Average # of Lanes per Node</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8.97</td>
<td>1.94</td>
<td>95.05</td>
<td>3.73</td>
<td>10.16</td>
</tr>
<tr>
<td>5</td>
<td>7.58</td>
<td>1.98</td>
<td>95.69</td>
<td>3.56</td>
<td>8.69</td>
</tr>
<tr>
<td>10</td>
<td>6.92</td>
<td>1.99</td>
<td>96.23</td>
<td>3.37</td>
<td>7.94</td>
</tr>
<tr>
<td># of Clusters</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7.81</td>
<td>1.97</td>
<td>95.69</td>
<td>3.54</td>
<td>8.90</td>
</tr>
<tr>
<td>10</td>
<td>7.83</td>
<td>1.97</td>
<td>95.63</td>
<td>3.57</td>
<td>8.97</td>
</tr>
<tr>
<td>Cluster Ratio</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>8.56</td>
<td>1.98</td>
<td>95.11</td>
<td>4.00</td>
<td>9.99</td>
</tr>
<tr>
<td>0.3</td>
<td>7.84</td>
<td>1.96</td>
<td>95.82</td>
<td>3.39</td>
<td>8.91</td>
</tr>
<tr>
<td>0.6</td>
<td>7.07</td>
<td>1.97</td>
<td>96.05</td>
<td>3.27</td>
<td>7.90</td>
</tr>
<tr>
<td>SC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>4.52</td>
<td>1.98</td>
<td>96.07</td>
<td>3.03</td>
<td>4.85</td>
</tr>
<tr>
<td>Yes</td>
<td>11.12</td>
<td>1.96</td>
<td>95.25</td>
<td>4.08</td>
<td>13.01</td>
</tr>
<tr>
<td>Bounds</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max</td>
<td>29.85</td>
<td>2.00</td>
<td>99.43</td>
<td>14.95</td>
<td>42.55</td>
</tr>
<tr>
<td>Min</td>
<td>0.57</td>
<td>1.73</td>
<td>82.82</td>
<td>0.50</td>
<td>0.58</td>
</tr>
</tbody>
</table>
The values in Table 2 presents the averages with respect to the different parameter settings of budget recovery percentage ("PBBR60-PBBR90") and instability percentage ("PBST90-PBST60") when the shippers are guaranteed a positive benefit by being allocated at most 60 to 90% of their stand alone costs.

Table 2  Average values of budget recovery and instability percentages with guaranteed positive benefits.

<table>
<thead>
<tr>
<th></th>
<th>PBBR60</th>
<th>PBBR70</th>
<th>PBBR80</th>
<th>PBBR90</th>
<th>PBST90</th>
<th>PBST80</th>
<th>PBST70</th>
<th>PBST60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>θ</td>
<td>0.25</td>
<td>72.44</td>
<td>93.33</td>
<td>97.85</td>
<td>99.39</td>
<td>0.79</td>
<td>2.36</td>
<td>INF</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>84.13</td>
<td>95.62</td>
<td>98.18</td>
<td>99.42</td>
<td>0.74</td>
<td>2.39</td>
<td>2.37</td>
</tr>
<tr>
<td></td>
<td>0.75</td>
<td>92.94</td>
<td>96.52</td>
<td>98.31</td>
<td>99.43</td>
<td>0.73</td>
<td>2.37</td>
<td>3.70</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>94.10</td>
<td>96.69</td>
<td>98.33</td>
<td>99.43</td>
<td>0.73</td>
<td>2.31</td>
<td>4.53</td>
</tr>
<tr>
<td># of Nodes</td>
<td>50</td>
<td>82.65</td>
<td>90.79</td>
<td>96.18</td>
<td>98.73</td>
<td>1.82</td>
<td>5.21</td>
<td>7.09</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>85.67</td>
<td>93.34</td>
<td>97.93</td>
<td>99.49</td>
<td>0.58</td>
<td>1.91</td>
<td>4.33</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>86.73</td>
<td>93.96</td>
<td>98.25</td>
<td>99.55</td>
<td>0.51</td>
<td>1.62</td>
<td>3.30</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>88.55</td>
<td>95.25</td>
<td>99.04</td>
<td>99.81</td>
<td>0.23</td>
<td>0.78</td>
<td>1.69</td>
</tr>
<tr>
<td>Average # of Lanes</td>
<td>2</td>
<td>85.16</td>
<td>92.85</td>
<td>97.55</td>
<td>99.32</td>
<td>0.88</td>
<td>2.61</td>
<td>4.49</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>86.07</td>
<td>93.46</td>
<td>97.93</td>
<td>99.42</td>
<td>0.74</td>
<td>2.26</td>
<td>3.71</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>86.47</td>
<td>93.69</td>
<td>98.07</td>
<td>99.44</td>
<td>0.74</td>
<td>2.20</td>
<td>3.37</td>
</tr>
<tr>
<td># of Clusters</td>
<td>5</td>
<td>85.92</td>
<td>93.35</td>
<td>97.86</td>
<td>99.38</td>
<td>0.80</td>
<td>2.27</td>
<td>3.61</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>85.89</td>
<td>93.32</td>
<td>97.84</td>
<td>99.41</td>
<td>0.77</td>
<td>2.44</td>
<td>4.05</td>
</tr>
<tr>
<td>Cluster Ratio</td>
<td>0</td>
<td>85.30</td>
<td>92.87</td>
<td>97.54</td>
<td>99.28</td>
<td>1.04</td>
<td>2.92</td>
<td>4.04</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>85.92</td>
<td>93.39</td>
<td>97.91</td>
<td>99.43</td>
<td>0.69</td>
<td>2.24</td>
<td>3.89</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>86.48</td>
<td>93.74</td>
<td>98.09</td>
<td>99.47</td>
<td>0.63</td>
<td>1.92</td>
<td>3.59</td>
</tr>
<tr>
<td>SC</td>
<td>No</td>
<td>88.19</td>
<td>94.87</td>
<td>98.80</td>
<td>99.68</td>
<td>0.36</td>
<td>1.13</td>
<td>2.30</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>83.61</td>
<td>91.79</td>
<td>96.90</td>
<td>99.10</td>
<td>1.21</td>
<td>3.60</td>
<td>6.14</td>
</tr>
<tr>
<td>Bounds</td>
<td>Max</td>
<td>99.18</td>
<td>99.66</td>
<td>99.87</td>
<td>99.97</td>
<td>6.55</td>
<td>17.77</td>
<td>35.21</td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>67.79</td>
<td>79.09</td>
<td>90.38</td>
<td>96.67</td>
<td>0.03</td>
<td>0.14</td>
<td>0.35</td>
</tr>
</tbody>
</table>

Our computational experiments show that the ratio of deadhead cost to the total cost is less than 7.9% on average and the maximum value for DH% is approximately 30% and belongs to a supply chain instance. Since there is a lane imbalance in SC instances, it is much more difficult to find a complementary lane to reduce repositioning. The range is approximately 1.97 on average, which is very close to the theoretical bound 2 (although the minimum is 1.73). This shows that, the core does not contain differentiated solutions especially for large instances.

For the game with the minimum liability restriction, on average 95% of the budget is recovered with a stable allocation (column MRBR), which shows that only 45% of the asset repositioning costs are accounted for. This is driven by the supply chain instances and instances with relatively small number of lanes. Also, percent instability value for Method SR (column MRSTMSR) is approximately 3.6% on average. Compared to the theoretical bound, $\phi_2 \cong 31.125\%$, this result is substantially better. Although the maximum value of this statistic is relatively large (15%), the minimum value is encouraging (0.5%). The percent instability value for the proportional cost allocation scheme currently used in the industry is 9% on average, with a maximum value of 43%
and a minimum value of 0.6%. These results empirically show that Method SR is superior to the proportional cost allocation scheme.

For allocations with positive benefits, approximately 86% of the budget is recovered when the allocated cost of any lane is at most 60% of its stand alone cost (column PBBR60). Note that, when $\theta = 0.25$, $\sigma = 0.6$ imposes the restriction that allocated cost of a lane should be at most 75% of the lane cost. Hence, these values are not surprising. As $\sigma$ is increased, the percentage of budget recovered also increases, first to 93% then to 98% and finally to 100%. Similarly, the percentage deviation from stability with $\sigma = 0.9$ (column PBST90) is on average 0.79% and slowly increases to approximately 5% as $\sigma$ decreases. Note that, some of the instances become infeasible with smaller $\sigma$ values and the infeasible instances are not included in calculating the averages, explaining why the maximum value when $\sigma = 0.7$ is greater than the maximum value when $\sigma = 0.6$.

The effect of different deadhead coefficients is not very significant on the key performance metrics. Note that, the total asset repositioning cost has a linear relationship with the deadhead coefficient. The budget recovery percentage decreases from 98% to 93% due to increase in the total asset repositioning cost. Also, the percent instability values for Method SR (columns MRSTMSR) for the allocation with the minimum liability restriction are approximately 2, 3 and 4 times the values when $\theta$ is 0.5, 0.75 and 1 respectively, which empirically suggests a linear relationship with $\theta$ as in the upper bound. This result is intuitive, in the sense that, for the allocations with the minimum liability restriction, the cost allocated among the members of the collaboration is the total asset repositioning cost since the shippers are already responsible for their original lane costs. The total asset repositioning cost has a linear relationship with the deadhead coefficient as stated above. This is also true for the proportional cost allocation method as can be seen in Table 1. Finally, percentage of budget recovered when $\sigma = 0.6$ (column PBBR60) increases from 72% to 94% since the constraint on allocating at most 60% of the stand alone cost to the shippers becomes less restrictive as $\theta$ increases.

Contrarily, we observe that the number of points is a significant factor on the performance indicators. The ratio of the deadhead cost to the total cost decreases substantially (from 11.7% to 4.5%) as number of points in the network increases. The budget recovery percentage increases from 94% to 97% due to decrease in the total asset repositioning cost. Also, the instability percentage for Method SR decreases from 5.1% to 2.3%. In general, we see that remaining performance measures improve steadily as number of points increases. This phenomenon can be explained as follows; as the number of points in the instances is increased, number of lanes to be covered is also increased in order to keep density of the network constant. Hence, the number of lanes with good
synergy increases. Also, as number of points increases, nodes become closer reducing the overall repositioning.

From both Tables 1 and 2, we see that average number of lanes per point is another factor that has a major impact on the results. The ratio of deadhead cost to overall cost decreases from 9% to 7% as average number of lanes per point increases from 2 to 10. Note that this reduction is generally seen in the non-supply chain instances. In the supply chain instances, increasing the average number of lanes does not increase synergy among the lanes due to restricted flow between nodes. In general, all the performance measures improve as number of lanes in the network increases, which suggests that as the collaboration expands the benefits from collaborating increase.

The significance of clusters is assessed by changing the fraction of points within a cluster. Note that, any non-cluster point is distributed uniformly across the total area so it may still fall within a cluster area. The values from both Table 1 and 2 show that increasing the cluster ratio improves the performance measures slightly as opposed to the insignificant affect of the number of clusters.

Finally, we can make the following observations regarding the results on the supply chain instances. The performance metrics for the supply chain instances are significantly worse on average. For example, the deadhead ratio increases to 11.1% from 4.5%. These results are expected since when a supply chain structure is present the requirement for repositioning is increased since a fraction of the points has no incoming arcs (suppliers) and a larger fraction of the points has no outgoing arcs (customers). This shows that repositioning charges increase when the network is mainly composed of vertically integrated supply chain elements, hence inter-firm collaboration becomes beneficial.

As stated before, we have tested the performance of our methods on real-life instances. The smallest of these instances has 15 nodes and 11 edges whereas the largest instance has 784 nodes and 5445 edges and the average number of edges per node is approximately 9.5 over all the instances. The performance indicators of the 18 real-life instances presents similar characteristics as the randomly generated instances, although from the values in Table 3, we see that real-life instances are more challenging due to the imbalance of loads. The interesting observation for these instances is the comparison between the percentage deviation from stability of Method SR and the proportional cost allocation scheme. The values in column INSPR are approximately 3 times of the values in columns MRSTMSR. Furthermore, on average for proportional cost allocation scheme the percent deviation from stability is approximately 25% over all the instances and all deadhead coefficient values which suggests that there exists a significant risk for the disintegration of the collaboration. For example, when \( \theta = 1 \), the deviation from stability is approximately 40%, which might be too
high of a value for depending on notions like “cost of collaborating” or “limited rationality” to keep the collaboration together.

6. Conclusion and further research directions

In this paper, we consider the cost allocation problem of a collaborative transportation procurement network and design effective and computationally efficient cost allocation methods to assist shippers to manage their collaboration structures. A good cost allocation mechanism should attract shippers to the collaboration, enable easier contractual agreements, and help to maintain the collaboration together. Although according to the CEO of Nistevo Network, Kevin Lynch, “The key to understanding collaborative logistics lies in recognizing how costs are distributed in a logistics network,” current practice only employs simple allocation mechanisms that may be very instable.

Our main concern is to develop implementable mechanisms to allocate gains from the collaboration in a stable manner to ensure the continuity of the collaboration. Due to several different challenges faced in establishing and managing a shippers’ logistics network, we identify several desirable cost allocation properties. However, we also show that no method can ensure allocations with all of these properties. As a result, we design several algorithms that generate allocations with worst case bounds on the relaxed properties.

Acknowledgments

Ozlem Ergun was partially supported under NSF grant DMI-0427446.

References


