TOBB UNIVERSITY OF ECONOMICS AND TECHNOLOGY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

COLLABORATIVE TRUCKLOAD TRANSPORTATION PROCUREMENT WITH MULTIPLE COALITIONS

M.Sc. THESIS

Soheyl ZEHTABIYAN

Department of Industrial Engineering

Supervisor: Assist. Prof. Dr. Gültekin KUYZU

April 2016

Approval of the Graduate School of Science and Technology.

………………………..
Prof. Dr. Osman EROĞUL
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

……………………….
Prof. Dr. Tahir HANALIOĞLU
Head of Department

I certify that I have read the thesis named as **"Collaborative truckload transportation procurement with multiple coalitions"** prepared by **Soheyl ZEHTABIYAN** and that in my opinion it is fully adequate, in scope and quality , as a dissertation for the degree of Master of Science.

Supervisor: **Assist. Prof. Dr. Gültekin KUYZU** ....
TOBB University of Economics and Technology

Examining Committee Members:

Assoc. Prof. Dr. Serhan DURAN (Chair) ....
Middle East Technical University

Assist. Prof. Dr. Salih TEKIN ....
TOBB University of Economics and Technology

THESIS STATEMENT

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

I hereby declare that all the information provided in this thesis was obtained with rules of ethical and academic conduct. I also declare that I have sited all sources used in this document, which is written according to the thesis format of Institute of Natural and Applied Sciences of TOBB ETU.

Soheyl ZEHTABIYAN

.

ABSTRACT

Master of Science

COLLABORATIVE TRUCKLOAD TRANSPORTATION PROCUREMENT

WITH MULTIPLE COALITIONS

Soheyl ZEHTABIYAN

TOBB University of Economics and Technology
Institute of Natural and Applied Sciences
Department of Industrial Engineering

Supervisor: Assist. Prof. Dr. Gültekin KUYZU

Date: April 2016

We study formation of stable coalitions given a set of shippers and their lanes corresponding to regularly scheduled truckload shipment. In this thesis, selecting participants, deciding who should participate with whom, calculating the lowest cost operational solution and allocating the system-wide cost to the participants stand out as important problems. Collaborating shippers try to identify tours which consist of regularly scheduled shipment with minimal empty truck movements. Then, they must allocate the total cost of the collaborative solution to the participated firms and individual lanes such that the collaborative solution remains attractive to the participants.

In the literature, solving the optimization problem minimization the total cost and allocating the calculated minimum cost are treated as successive but distinct phases. The cost minimizing optimization problem is solved with well-known operation research methods, while cooperative game theory concepts are used for cost allocation.

The minimum cost solution may render finding an acceptable cost allocation impossible. Besides, similar works in the literature assume that the collaborating firms will forge a single grand coalition. However, as the collaboration grows in size, a single grand coalition may become impractical and also it might leave several lanes out of the coalition, depriving shippers of significant cost savings.

In this study, we propose algorithm to design coalition structure which consist of multiple disjoint stable coalitions. Each coalition must have a minimum cost collaborative solution with an acceptable cost allocation. Due to the complexity of the task hand, we devised a heuristic to find good quality solutions to this problem.

Keywords: Lane covering, Collaborative logistics, Cooperative game theory, Coalition structure, Heuristics.

ÖZET

Yüksek Lisans Tezi

ÇOK KOALİSYONLU TAM KAMYON YÜKÜ GÖNDERİCİ İŞBİRLİĞİ

Soheyl ZEHTABIYAN

TOBB Ekonomi ve Teknoloji Üniveritesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Yrd. Doç. Dr. Gültekin KUYZU

Tarih: Nisan 2016

Gönderici işbirliği, son yıllarda ortaya çıkmış yeni bir işbirliği türüdür ve tedarik zinciri yönetiminde kurumlar arası yatay işbirliği sınıfına girmektedir. Sert rekabet koşulları, kaynak yetersizliği, iklim değişimi, güvenlik sorunları ve yeni kanuni düzenlemeler firmalar üzerindeki baskıyı artırmış ve geleneksel düşünce kalıplarını zorlayan yeni çözümler aramaya itmiştir. İşbirliği; daha geniş ve bütün sistemi kapsayan bir bakış açısı getirmesi nedeniyle yeni fırsatlar sunan bir strateji olarak görülmektedir. Gönderici işbirliğinde taşıyıcı firmalardan taşımacılık hizmeti alan bir grup gönderici firma bir araya gelir; ve taşıyıcı firmalarla grup olarak pazarlık yaparlar. Göndericiler işbirliği yapmak istediklerinde; hangi göndericilerin işbirliğine dahil edileceği, hangi göndericilerin rotalarının arka arkaya ekleneceği, ve oluşturulan rota birleştirme çözümünden doğan toplam maliyetin göndericilere ve hatta her bir rotaya dağıtılması konularında en iyi kararları vermek durumundadırlar.

Literatürdeki çalışmalar işbirliği yapan göndericilerin tek bir koalisyon kurduğunu varsaymış ve bu tek koalisyonunun kurulması, devamı ve genişletilmesi konularını ele almışlardır.

Verilen bir koalisyon ve bu koalisyon için hesaplanan en küçük maliyetli çözüm için adil bir maliyet paylaşımı olup olmadığını konu almışlardır. Buna ek olarak, verilen bir koalisyon, çözüm ve maliyet paylaşma mekanizması için koalisyona katılmak isteyen yeni bir göndericinin koalisyona alınıp alınmaması kararını konu alan çalışmalar da mevcuttur. Büyük ölçekli gönderici işbirliği ağlarında sadece tek bir koalisyona izin verilmesi koordinasyonu zorlaştırmakta ve kabul edilebilir maliyet dağıtımına sahip geniş çaplı bir çözüm bulunmasını zorlaştırmaktadır.

Bu çalışmada birden fazla ayrışık koalisyon içerebilen tam kamyon yükü gönderici işbirliği ağları ele alınmıştır. Her biri adil maliyet dağıtımına sahip, ayrık koalisyonlardan oluşan en düşük toplam maliyetli koalisyon yapısının bulunması amaçlanmıştır. Gönderici ve rota sayıları arttıkça ve operasyonel kısıtlar eklendikçe bu kararları en iyi biçimde vermek gittikçe zorlaşmaktadır. Gerçek hayat durumlarında problem boyutlarının çok büyük olması beklendiği için özellikle büyük ölçekli problem örneklerinin çözümüne yönelik sezgisel algoritmalar geliştirilmiştir.

Anahtar Kelimeler: İşbirlikçi lojistik ve tedarik zinciri, Sezgisel, Maliyet dağıtımı, İsbirlikliçi oyun kuramı

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to my supervisor Assist. Prof. Dr. Gültekin Kuyzu, not only for the unique chance he gave me to join his research team, but for all precious lessons, his patience, guidance and his comprehensive support to complete this thesis. The door to Dr. Kuyzu's office was always open whenever I ran into a trouble spot or had a question about my research, writing or other part of my academic life.

My gratitude goes to the Scientific and Technological Research Council of Turkey (TUBITAK) for their support provided as part of project No 112M227. I also thank TOBB University of Economics and Technology for providing me a scholarship throughout my study and to all staff members and assistance of the Industrial Engineering Departmant of TOBB University of Economics and Technology for their deep teaching, advises and suggestions, which I appreciate the most. I would like to thank my officemate Nihat Öner for all the joyful moments we spent on this project.

On more personal level, I should thank my patient and understanding girlfriend Mahsa for all these two years. She stuck with me during the long months of my study even when I retreated to long days with my computer. I am also very grateful to my kind sister, Sorour, for her energy, support and her love.

Finally, I would like to dedicate this thesis to my parents. For their support, encouragement, endless love and their sacrifices. I admire them for all of their accomplishments in life and for my education, as the greatest joy for them. I sincerely appreciate them for their indispensable role in my life and for all of the knowledge and wisdom that they have passed on to their children over the years.

CONTENTS

LIST OF TABLES

Page

1. INTRODUCTION

Nowadays, transportation service providers are under a forbidding pressure. Increasing pressure of customers to transportation service providers for better service and the extremely dynamic circumstances have made the companies find more efficient solutions for their operations. In order to overcome the ever growing operational issues, companies employ various methods for managing their freight transportation networks. Firms have found horizontal collaboration as an effective way to decrease their inefficiencies. Companies with traditionally separate supply chains increasingly seek ways to identify and exploit win-win situations in order to improve performance. In this study, we focus on collaborative truckload transportation procurement (CTTP), which is one of the forms of horizontal collaboration in the supply chain.

In CTTP, companies aim to eliminate their empty truck movements which is generally the main source of inefficiency. Consequently, a group of shippers purchasing the service of carriers can negotiate with carriers for better rates. More specifically, collaborating shippers offer the carriers regularly scheduled multi-company tours with little repositioning in exchange for reduced rates. These tours are attractive to the carriers because reduced repositioning recovers lost operational efficiency due to empty truck movements, and the regularity of the schedules help with driver retention, another major concern for the carriers. We can take Transplace and Nistevo companies in US and also Schenker and Celexor in Europe as some examples for companies which have used this method to reduce their transportation costs.

After forming a coalition, shippers will know their partners and they can determine their pre-specified route. Then, they can determine the synergy of their collaboration by measuring the total cost saving. Allocating the total cost among the shippers will play a crucial role in maintenance of the obtained coalition. The departure of any of the coalition members could affect the whole system and sometimes it could even collapse the grand coalition. That is why designing a fair cost allocation method is one of the indispensable topics in the CTTP literature.

In the literature on collaborative logistics, participants are assumed to be given, and solving the optimization problem which minimizes the total cost and allocating the calculated minimum cost are treated as successive phases. The method of selecting the participants is typically left unspecified. The cost minimizing optimization problem is solved with well-known operation research methods, while cooperative game theory concepts are used for cost allocation. The minimum cost collaborative solution may render finding a fair cost allocation impossible.

In order to obtain a single stable coalition, a great number of the lanes may have to be excluded from the collaboration, which would not yield a favorable result in terms of the savings of the shippers. Due to this fact, not only forming a single coalition would be impractical in real-world problems, but also it would cause a significant increase in the cost of collaboration.

In the literature, as the first step, determining a grand coalition with minimum cost, has been studied by various operation research methods. Then, they have used the obtained solution in the first step to suggest a fair cost allocation. In order to allocate total minimized cost fairly, cooperative game theory concepts have been used frequently as a benchmark in the second step. According to the cooperative game theory, a fair cost allocation has three important characteristics:

- *Budget balance*: In a budget balanced (efficient) system, the total amount of allocated cost among the players must be exactly equal to the whole system's cost. According to this feature, no deficit or surplus are allowed in a budget balanced system.
- *Stability:* In a stable system, none of the collaboration's elements can find a better alternative, so they would prefer to stay rather than breaking away from coalition.
- *Individual rationality:* In an individual rationally system, each player should not gain less profit after entering a coalition, compared to its standalone cost.

Note that, in cooperative game theory, if a cost allocation can satisfy stability and budget balance, it would be in the core. However the core of a cooperative game might be empty. In fact, the cores of the cost allocation games arising in CTTP are highly probable to be empty, under practical circumstances.

It is worth to mention that similar works in the literature assume that the collaborating firms will form a single grand coalition. However, due to the great number of shippers in the real-world problems, it is more likely that shippers will form multiple coalitions.

As a brief look, we can categorize some major cooperative game theory based methods in the literature which are struggling to find an acceptable solution:

- *Shapley Value*: One of the most widely known cost allocation methods in cooperative game theory is the Shapley value [30]. The Shapley value for a player is the weighted average of the player's marginal contribution to each subset of the collaboration. It can be interpreted as the average contribution of each player if the coalition was built by adding one player at a time [5]. Unfortunately, a non-empty core may not include the cost allocation obtained by the Shapley value. Another major drawback of using the Shapley value in collaborative logistics is that it requires the solution of an exponential number of optimization problems due to the need to calculate the marginal contribution of each customer to each subset.

- *Nucleolus*: The nucleolus [29] is another well known cost allocation concept. The nucleolus is the unique cost allocation that lexicographically maximizes the minimal gain over all of the subsets of the collaboration. The nucleolus is in the core if the core is non-empty. It is considered to be more desirable than others when there are multiple cost allocations in the core. The nucleolus may exist even when the core is empty.

- *Dual based methods*: An alternative cost allocation approach is the use of dual prices when the cost or gain of each coalition is given by a linear program. In fact, linear programming duality has been used to build core and approximate core allocations for various cooperative games. In collaborative logistics, the cost of a coalition is usually calculated by solving a covering type integer program. A classic theorem by Bondareva [6] and Shapley [31] implies that for covering games in which the cost of a coalition (or sub-coalition) is given by the minimum cost solution to a covering integer program, the core is nonempty if and only if the linear relaxation of the cost calculation IP has no integrality gap [28]. Thus, constructing the coalition in a way to maximize the total savings will result in an unstable coalition in most cases of logistics collaborations. Despite the fact that this method has been used frequently in similar studies, calculating the linear programing and also dual solutions for large real-word problems is extremely time consuming and impractical.

In CTTP, the whole system's repositioning cost can be minimized by solving a lane covering problem (LCP). LCP is a covering problem which can be solved efficiently as minimum cost circulation flow problem which determines the order of lanes in a minimized cost cycle. Even though LCP is an extremely time efficient method, it lacks important practical constraints on the cycles, e.g., length of the tours which cover the lanes or the maximum number of the partners each shipper can collaborate. Adding such constraints to the LCP results in NP-Hard LCP variants [10, 11, 21]. These constraints also complicate the cost allocation process. While the base LCP yields a non-empty core [26], constrained LCP variants are very highly likely to yield an empty core. As an alternative, we can eliminate some elements of the grand coalition to obtain a coalition with a fair cost allocation at the expense of increasing the total system-wide cost. Furthermore, we can optimize this trade-off by solving a MIP [25], but this approach may leave a high number of lanes outside of the collaboration.

As mentioned before, we follow an integrated approach in this study. Given a set of candidate shippers and lanes belonging to these shippers, our objective is to find the minimum cost coalition structure such that each coalition in the structure ensures the existence of a fair cost allocation. In this thesis, we assume that multiple coalitions can be formed in the CTTP network. We focus on determining a set of CTTP coalitions such that each coalition is *core stable*, i.e. has a non-empty core, and the sum of the total cost values of the coalitions is minimized. We assume that the cost of each coalition is obtained by solving an NP-Hard LCP variant. We extend the MIP formulation of Öner and Kuyzu [25] to the case of multiple coalitions. Since solving this MIP requires embedding column and row generation into the branch-and-bound tree, we propose a heuristic approach based on repeatedly solving unconstrained LCPs, which can be used to obtain good quality solutions efficiently.

In this study, we first provide a formal statement of the problem we are working on (§3). Second, we provide a mathematical model and analyze the difficulty of solving the model. Then, we present our heuristic solution approach (§4). In §5, we demonstrate how our approach performs through computational experiments on randomly generated instances, and we provide concluding remarks in §6.

2. LITERATURE REVIEW

As there are a large number of possible areas of supply chain management in which collaboration can be formed, we see a great variety of studies on collaborative networks in logistics and supply chain (see Danloup et al. [9]). Collaboration is widespread in logistics and supply chain: ranging from purchasing, demand planning, and inventory management to warehousing and transportation. Erhun and Keskinocak [12] review various forms of collaboration in the supply chain, and outline the potential benefits of each.

Collaborative logistics has received an increasing level of attention in recent years. Audy et al. [3] present five different coordination mechanisms found in the literature to support collaborative logistics, which the authors differentiate by their planning function, sharing approach, and information, decision and financial flows. It is noted that, in most cases, the logistics solution is planned first and then the sharing is set based on the plan. The authors list a very few number of works in which logistics planning and gain sharing is addressed simultaneously: most notably Agarwal and Ergun [1] who design a mechanism to motivate the carriers in a liner shipping alliance to act in the best interest of the alliance while maximizing their own profits.

Note that the research presented in this thesis approaches collaborative logistics from the perspective of the customers of the carriers. Extensive collaborative activities among carriers (including 3PLs) have been in place for a long period of time. There is a significant body of research studying collaborative logistics from the perspective of carriers. Most of the studies in this category, try to propose efficient ways to reduce the cost of carriers, based on the same concepts (capacity sharing, decreasing operational cost,...). Hernández et al. [16] studied collaboration problem under dynamic capacities. Kwon et al. [22] propose integrated multi-round mechanism for truckload transportation. Similar to our study, they provide a mathematical model and heuristic algorithm to their problem. Agarwal and Ergun [1] and Kuo and Miller-Hooks [20], study other problem variants of Carrier collaboration in the literature and also, Hu et al. [18], Agarwal et al. [2], Kuo and Miller-Hooks [20], and Özener et al. [27] provide overviews of collaboration among carriers in air, ocean, rail, and road transportation, respectively.

Since collaboration among the shippers is a relatively new practice in the literature of collaborative logistics, unlike the carrier collaboration, there are few studies related to shipper collaboration. Moore Jr et al. [24] used optimization. and simulation models together in order to find the efficient routs of carriers for a centralized transportation procurement department at Reynolds Metal Company. Cruijssen et al. [8] report a cooperative distribution agreement reached in 1993 between eight competing medium-sized Dutch producers of sweet and candy, and provide a broad review of horizontal cooperation in transport and logistics. Groothedde [14] solves a hub network design optimization model using a simulated annealing algorithm which sequentially adds shippers to gradually build the collaborative hub network. Cruijssen et al. [7] study joint route planning among companies with distinct distribution networks but whose truck depots are sufficiently close. In the last two works, the underlying optimization problems solved are well known in the vehicle routing literature, and the Shapley Value is the chosen cost allocation method. Ergun, Kuyzu, and Savelsbergh [10, 11] study the optimization problems arising in collaborative truckload transportation procurement networks managed by 3PL companies in the U.S. They study the minimization of continuous move tours which cover the given sets of lanes and also propose the using of LCP as well as greedy heuristics for some problems with driver and time restrictions (NP-hard). Immorlica et al. [19] study cycle covering problem for bounded size cycles which cover an edges subset of a graph. They give a (1+ln 2)-approximation for the cardinality constrained LCP. Kuyzu [21] introduces another constraint into the LCP, motivated by the need to limit the number of partners with whom the collaborative tours must be coordinated, develops a column generation approach for the solution of the resulting NP-Hard LCP variant. Özener and Ergun [26] study the cost/benefit allocation problem of the LCP, derived from cooperative game theory. They design allocation methods based on optimal dual prices to maintain the collaboration among selfish collaborators. Hezarkhani et al. [17] further characterize the theoretical properties of dual based methods for allocating the cost of the LCP. They put forward a general framework for the possibilities/impossibilities of a complete characterization of the core in lane covering games, which is obtained by dual solutions. Öner and Kuyzu [25] develop row and column generation based approaches for the selection of a single core stable coalition in CTTP when the operational cost is obtained by solving an LCP with cardinality, length, and/or parter constraints.

Frisk et al. [13] suggest the equal profit method (EPM) which relatively is a new way of allocating costs. They study cost allocation methods for a large fleet sharing application in southern Sweden with eight forest companies. According to them, EPM is a more favorable method to coalition members than other well-known methods, because it has several implicit advantages for maintaining the collaboration. They claim that the cost allocation obtained by the EPM is shown to be in the core when it is not empty. Audy et al. [4] modify EPM and the alternative Cost Allocation methods to allocate costs among four Canadian furniture companies. Lozano et al. [23] study the advantage of collaboration in transportation problems. They compare different cooperative game theory concepts for allocating benefits of horizontal cooperation by enumerating all of the possible coalitions among four different companies under a simple linear cost saving model.

All of the above mentioned works assume that the partners and the operational solution of the collaboration are determined independent of the cost allocation. A few works have recently relaxed this assumption. Vanovermeire and Sorensen [33] devise methods for measuring and rewarding flexibility in collaborative logistics and incorporate them into existing cost allocation mechanisms, such as the Shapley value and the nucleolus. They propose a cost minimization problem which is yielded by the integration of the operational planning methods in to the mentioned well-known cost allocation methods. Vanovermeire and Sorensen [32] develop an iterative coalition building heuristic that integrates the Shapley value and due date change penalties into the cost-minimization problem in a collaborative setting involving the synchronized consolidation of transportation orders. The method is tested on single origin-destination instances with three collaborating companies.

To the best of our knowledge, Guajardo and Rönnqvist [15] are the first to consider multiple coalitions in a collaborative logistics setting. They point out some practical issues related to the sustainability of giant coalitions and propose mixed integer linear programming models to partition the participants into multiple coalitions given a set of candidate coalitions and their associated costs, in order to obtain core stability and strong equilibrium in smaller coalitions. They assume that a third party, such as a team of consultants suggests the set of candidate coalitions and a cost minimizing way to implement the collaboration among the companies in each candidate coalition. Note that these sub-coalitions are not necessarily formed to guarantee a non-empty core. The models are tested on a collaboration among eight forest companies in the form of backhauling and wood bartering.

This study extends the existing literature on CTTP by integrating coalition member selection, operational solution identification, and cost allocation where the underlying optimization problem is a NP-Hard constrained LCP variant. This is also one of first works considering selection of coalition members, and/or formation of multiple coalitions in horizontal logistics collaboration.

3. PROBLEM DEFINITION

Given a set of shippers who each has identified a set of lanes (i.e. regularly scheduled truckload movements) as candidates for collaborative procurement, our objective is to identify the least costly way of dividing the shippers and their lanes into coalitions such that the lanes included in each coalition are covered by tours with minimal empty truck movements and the cost of each coalition is distributed to the lanes in an acceptable manner. Each shipper wants to share tours with a limited number of partners.

More formally, given a complete graph $G = (N, A)$ with node set N, arc set A, non-negative arc lengths $f_a\ \forall a \in A$, a set of shippers P, and a set of lanes $L_i \subseteq A$ from each shipper $i \in P$ (note: $L_i \cap L_j = \emptyset$ for $i \neq j$), the objective is to find a set of directed simple cycles covering the lanes in $L = \bigcup_i L_i$ with minimum total cost such that:

- C1 The union of the cycles which cover the lanes of L_i includes lanes from at most MSP_i other shippers.
- C2 Each lane is included in at most one coalition.
- C3 The lanes of L are divided into at most MK coalitions, and each coalition is core stable.
- C4 The lanes of L_i are included in at most MSK_i coalitions.
- C5 Each coalition includes at most MKL lanes.
- C6 The sizes of any two coalitions do not differ by more than $MKLD$ lanes.

We assume that the cost of travel is proportional to the distance traveled. The empty movement cost of a truck is expected to be lower than its loaded movement cost because of reduced fuel consumption and less equipment wear. Therefore, we compute the cost of repositioning along an arc $a \in A$ by multiplying the arc length f_a by a repositioning coefficient $\rho_a \in (0, 1)$. We take the cost of traversing a lane with a loaded truck as simply equal to the length of the lane. The cost of a cycle is the sum of the costs of its lane arcs and repositioning arcs.

3.1 Core Stability in Constrained Lane Covering

As mentioned before, we are looking for a set of coalitions which consist of regularly scheduled truckload movements with the least possible empty movement. In other words, the lanes entered in the coalitions should be covered by cycles with the least repositioning amount. Besides, recall that we want each coalition to be core stable. We model the cost allocation problem as a cooperative game and allocate a cost of w_l to each lane $l \in L$, similar to Özener and Ergun [26]. Let $F(S)$ be the cost of lane set $S \subseteq L$, which is calculated by solving the LCP on S constrained by C1. This NP-Hard LCP variant is the partner constrained LCP (PCLCP), which can be solved by a set covering/partitioning formulation over the set of all feasible cycles that can be used to cover the lanes in S [21].

Let the cost allocation game defined by coalition $L_{sub} \subseteq L$ (participated lanes in coalition) and cost function F be denoted as $\langle L_{sub}, F \rangle$ for each coalition. A cost allocation $w_l, l \in L_{sub}$ for PCLCP is in the core of $\langle L_{sub}, F \rangle$ if and only if it satisfies the following:

$$\sum_{l \in L_{sub}} w_l = F(L_{sub}) \tag{3.1}$$

$$\sum_{l \in S} w_l \leq F(S) \qquad \forall S \subseteq L_{sub} \tag{3.2}$$

However, we can simplify the above conditions with the help of the following theorem:

Theorem 1 *Replacing inequalities (3.2) with inequalities*

$$\sum_{l \in c} w_l \leq F(c) \qquad \forall c \in C_{sub} \tag{3.3}$$

will not change the set of cost allocations in the core.

Proof: This proof is based on the fact that the cost $F(S)$ of each subset S is equal to either the cost of a cycle or the sum of the costs of a set of cycles covering S. Let C_{sub} be the set of all feasible cycles (under C1), which can be used to cover the lanes in L_{sub}. Hence, we can claim that since each cycle corresponds to a set of lanes, a cost allocation $w_l, l \in L_{sub}$ satisfying (3.2) automatically satisfies (3.3). Furthermore, the cost of each subset of lanes is the sum of the costs of a set of feasible cycles. Consequently, if the allocated costs satisfy (3.3), they will also satisfy (3.2). □

The PCLCP can be formulated as a set covering problem or a set partitioning problem over the entire set of all feasible cycles (under C1). Let $s_{cl} \in \{0,1\}$ be a parameter indicating whether each lane $l \in L_{sub}$ is included in cycle $c \in C_{sub}$. Let the parameter f_c represent the cost of each feasible cycle $c \in C_{sub}$. For each feasible cycle $c \in C_{sub}$ and shipper company pair $\{i, j\} \subseteq P$, define a parameter p_{cij} which denotes if the cycle leads to a common collaborative tour between the shipper pair. As stated in the definition of the problem, let MSP_i indicate the maximum number of partners that each shipper i is willing to accept. Let $M_{ij} = \min\{|L_i|, |L_j|\}$. Define binary decision variables x_c to indicate whether a feasible cycle $c \in C_{sub}$ is selected in the optimal solution or not. For each shipper pair $\{i, j\} \subseteq P$, define binary decision variables y_{ij} to indicate whether these two shippers share a collaborative tour in the optimal solution. The resulting set partitioning formulation is below.

$$\text{SPP:} \quad \min \quad \sum_{c\in C_{sub}} f_c x_c \tag{3.4}$$

$$\text{s.t.} \quad \sum_{c\in C_{sub}} s_{cl} x_c = 1 \qquad \forall l \in L_{sub} \tag{3.5}$$

$$M_{ij} y_{ij} - \sum_{c\in C_{sub}} p_{cij} x_c \geq 0 \qquad \forall \{i,j\} \subseteq P \tag{3.6}$$

$$-y_{ij} + \sum_{c\in C_{sub}} p_{cij} x_c \geq 0 \qquad \forall \{i,j\} \subseteq P \tag{3.7}$$

$$-\sum_{j\in P\setminus\{i\}} y_{ij} \geq -MSP_i \qquad \forall i \in P \tag{3.8}$$

$$x_c \in \{0,1\} \qquad \forall c \in C_{sub} \tag{3.9}$$

$$y_{ij} \in \{0,1\} \qquad \forall \{i,j\} \subseteq P \tag{3.10}$$

In the formulation, constraints (3.5) ensure that every lane is covered by exactly one feasible cycle. Constraints (3.6) force the value of the decision variable y_{ij} to be equal to 1 if the solution has at least one feasible cycle which contains lanes from both shippers i and j. Conversely, constraints (3.7) ensure that the value of decision variable y_{ij} is equal to 0 if the solution has no feasible cycles containing lanes from both of shippers i and j. Constraints (3.8) make sure that each shipper i shares feasible cycles with at most MSP_i other shippers. Constraints (3.9) and (3.10) are the integrality constraints.

Theorem 2 *The core of $\langle L_{sub}, F\rangle$ is non-empty if and only if the set covering / partitioning formulation of the PCLCP has zero integrality gap and its constraints (3.6)-(3.8) are redundant.*

Proof: Equations (3.3) can be interpreted as the feasible region of the dual of the linear relaxation of SPP without constraints (3.6)-(3.8). In this case, the allocated costs w_l correspond to the dual variables. Let $H(L_{sub})$ be the optimal objective value of the following linear program (over the lane set L_{sub}):

$$\text{SPD:} \quad \max \quad \sum_{l\in L_{sub}} w_l \tag{3.11}$$

$$\text{s.t.} \quad \sum_{l\in c} w_l \leq f_c \qquad \forall c \in C_{sub} \tag{3.12}$$

$$w_l \in \mathbb{R} \qquad \forall l \in L_{sub} \tag{3.13}$$

Let $w_l^{SPD}, l \in L_{sub}$ comprise an optimal solution of the linear program SPD. Clearly, $H(L_{sub}) = \sum_{l\in L_{sub}} w_l^{SPD}$. Because of constraints (3.6)-(3.8) combined with integrality of the decision variables and by well known duality theorems, we have $\sum_{l\in L_{sub}} w_l \leq$

$H(L_{sub}) \leq F(L_{sub})$ for any cost allocation satisfying the stability conditions. Furthermore, $\sum_{l \in L_{sub}} w_l^{SPD} = F(L_{sub})$ if and only if SPP has a zero integrality gap and the constraints (3.6)-(3.8) are redundant. Hence, we cannot find a cost allocation that is both budget-balanced and stable unless the LP relaxation of SPP without constraints (3.6)-(3.8) is optimal for the full SPP. It follows that the core is non-empty if and only if SPP has a zero integrality gap and its constraints (3.6)-(3.8) are redundant. This result is in line with a classic theorem on cost allocation games with set covering type cost functions [6, 28]. □

This proof also implies another consequence, which provides the base of our study. **For an arbitrary selected set of lanes L, keeping all of the lanes of L together as a single core stable coalition is not likely.** In this study, we devise an effective algorithm to provide stable coalitions. We will discuss further details of our algorithm in the upcoming sections.

3.2 Multiple Core Stable Coalitions Selection

The discussed formulation (SPP) is the most basic form of our problem which does not provide lane selection and cost allocation decisions. We can formulate an integer programming model which integrates mentioned properties and stability conditions into the set partitioning formulation of the PCLCP.

We incorporate additional desirable cost allocation properties with the help of this model. We place an upper bound on each allocated cost to guarantee a certain percentage of savings for each lane in the coalition. We also place a lower bound on each allocated cost to prevent free-riders in the coalition.

Let C be the set of all feasible cycles. Let $s_{cl} \in \{0,1\}$ be a parameter indicating whether each lane $l \in L$ is included in cycle $c \in C$. Let the parameter f_c represent the cost of each feasible cycle $c \in C$, i.e. $f_c = F(c)$. Let g_l be the cost of lane $l \in L$ outside the collaboration. For each feasible cycle $c \in C$ and shipper company pair $\{i, j\} \subseteq P$, define a parameter p_{cij} which denotes if the cycle leads to a common collaborative tour between the shipper pair. Similar to the SPP, let $M_{ij} = \min\{|L_i|, |L_j|\}$. Let $K = \{1, .., MK\}$ be the set of coalitions. Let λ_l be the lower bound of the cost allocated to lane l in the collaboration. Let θ_l be the minimum proportional savings for lane $l \in L$ if included in the collaboration.

We first define binary decision variables x_c to indicate whether a feasible cycle $c \in C$ is selected in the optimal solution or not. For each shipper pair $\{i, j\} \subseteq P$, we define binary decision variables y_{ij} to indicate whether these two shippers share a collaborative tour in the optimal solution.

Let u_{lk} be the binary decision variable indicating whether lane l is included in coalition k from the collaboration. Let U_l be the binary decision variable indicating whether lane l is excluded from all of the coalitions. Let w_{lk} be the decision variable corresponding to the cost allocated to lane l in coalition k. Our multiple coalition selection integer program (MCSIP) is formulated as follows:

MCSIP:

$$\min \quad \sum_{c \in C} f_c x_c + \sum_{l \in L} g_l U_l \tag{3.14}$$

$$\text{s.t.} \quad \sum_{c \in C} s_{cl} x_c = 1 - U_l \qquad \forall l \in L \tag{3.15}$$

$$w_{lk} \leq (1 - \theta_l) g_l u_{lk} \qquad \forall l \in L, k \in K \tag{3.16}$$

$$w_{lk} \geq \lambda_l u_{lk} \qquad \forall l \in L, k \in K \tag{3.17}$$

$$\sum_{c \in C} p_{cij} x_c \leq M_{ij} y_{ij} \qquad \forall \{i, j\} \subseteq P \tag{3.18}$$

$$\sum_{c \in C} p_{cij} x_c \geq y_{ij} \qquad \forall \{i, j\} \subseteq P \tag{3.19}$$

$$\sum_{j \in P \setminus \{i\}} y_{ij} \leq MSP_i \qquad \forall i \in P \tag{3.20}$$

$$\sum_{k\in K} u_{lk} + U_l = 1 \qquad \forall l \in L \qquad (3.21)$$

$$x_c + (u_{l_1k} - u_{l_2k}) \leq 1 \qquad \forall c \in C, \{l_1, l_2\} \subseteq L_c, k \in K \qquad (3.22)$$

$$x_c - (u_{l_1k} - u_{l_2k}) \leq 1 \qquad \forall c \in C, \{l_1, l_2\} \subseteq L_c, k \in K \qquad (3.23)$$

$$v_{ik} \geq u_{lk} \qquad \forall i \in P, l \in L_i \quad k \in K \qquad (3.24)$$

$$v_{ik} \leq \sum_{l\in L_i} u_{lk} \qquad \forall i \in P, k \in K \qquad (3.25)$$

$$\sum_{k\in K} v_{ik} \leq MSK_i \qquad \forall i \in P \qquad (3.26)$$

$$\sum_{k\in K} u_{lk} \leq MKL \qquad \forall l \in L \qquad (3.27)$$

$$\sum_{i\in P} v_{ik} \leq MKS \qquad \forall k \in K \qquad (3.28)$$

$$\sum_{i\in P} u_{i\,k_1} - \sum_{i\in P} u_{i\,k_2} \leq MKLD \qquad \forall k_1, k_2 \in K : k_1 \neq k_2 \qquad (3.29)$$

$$\sum_{l\in L}\sum_{k\in K} w_{lk} - \sum_{c\in C} f_c x_c = 0 \qquad (3.30)$$

$$\sum_{l\in c} w_{lk} \leq f_c \qquad \forall c \in C, k \in K \qquad (3.31)$$

$$w_{lk} \leq g_l u_{lk} \qquad \forall l \in L, k \in K \qquad (3.32)$$

$$x_c \in \{0,1\} \qquad \forall c \in C \qquad (3.33)$$

$$y_{ij} \in \{0,1\} \qquad \forall \{i,j\} \subseteq P \qquad (3.34)$$

$$u_{lk} \in \{0,1\} \qquad \forall l \in L, k \in K \qquad (3.35)$$

$$U_l \in \{0,1\} \qquad \forall l \in L \qquad (3.36)$$

$$v_{ik} \in \{0,1\} \qquad \forall i \in P, k \in K \qquad (3.37)$$

$$w_{lk} \geq 0 \qquad \forall l \in L, k \in K \qquad (3.38)$$

In the formulation above, the objective (3.14) minimizes the sum of the total cycle cost and the total uncooperative cost. Constraints (3.15) ensure that every lane included in a coalition is covered by exactly one cycle. Constraints (3.16) satisfy the individual rationality. This constraint ensure that every lane included in a coalition obtains specific percentage savings. Constraint (3.17) provides a lower bound for each lane's allocated cost (otherwise, since the MCSIP is a minimization problem, the optimal value of he allocated cost would be equal to zero). Constraints (3.18) force the value of the decision variable y_{ij} to be equal to 1 if the solution has at least one cycle which contains lanes from both shippers i and j. Conversely, constraints (3.19) ensure that the value of decision variable y_{ij} is equal to 0 if the solution has no cycles containing lanes from both of shippers i and j. Constraints (3.20) make sure that each shipper i shares cycles with at most MSP_i other shippers. Constraints (3.21) require that each lane is included in at most one coalition. Constraints (3.22) and (3.23) make sure that the lanes of a selected cycle are in the same coalition. Constraints (3.24) and (3.25) establish consistency between the lanes and the shippers included in each coalition. Constraints (3.26), (3.27), and (3.29) satisfy conditions C4, C5, and C6, respectively. Constraints (3.30) and (3.31) are the budget balance and stability constraints, respectively. Constraints (3.32) ensure that every lane included in a coalition obtains non-negative savings, and the cost allocated to lane l from coalition k is zero if the lane is excluded from that coalition. Constraints (3.33)-(3.38) are integrality and non-negativity constraints.

Note that any lane subset $S \subseteq L$ will either be a cycle or its cost $F(S)$ will be sum of the costs of a set of cycles. Each w_{lk} can be positive for at most one coalition k, and constraints (3.24) and (3.25) ensure that all of the lanes of a selected cycle are in the same coalition. Since $w_{lk} = 0$ for coalitions which do not include lane l, a constraint in (3.31) will be redundant unless all of the lanes of the corresponding cycle are in the same coalition. Constraints (3.31) ensure that each coalition is stable within itself. An implication of these constraints is that the total cost allocated to the lanes of a selected cycle is equal to the cost of the cycle. Therefore, the total cost allocated to the lanes of each coalition will be equal to its cost, which means that constraints (3.30) is sufficient for achieving budget-balance within each selected coalition. We conclude that constraints (3.30) and (3.31) are sufficient for ensuring core stability of each selected coalition.

Since we formulate the MCSIP over the set of all feasible cycles, the real-world large instances requires column generation and advanced integer programming solution techniques such as branch-and-price. For instance, constraints (3.22), (3.23), and (3.31) are defined over the set of feasible cycles, which necessitates embedding row generation along with column generation in the branch-and-bound procedure. The effectiveness of this exact solution approach is expected to diminish rapidly as the problem size grows even for the case of $MK = 1$, which can be formulated without several of the constraints in the MCSIP [25]. Hence, this formulation is not a preferable method for large scale real world problems. Due to the exponential numbers of rows and columns and complexity of the task hand, we opt for a heuristic algorithm. We provide the details of our solution algorithm in the next section.

4. SOLUTION APPROACH

Our solution approach is based on the observation that the cost allocation game defined by the base LCP has a non-empty core [26]. We aim to select the coalitions such that solving the base LCP on each coalition results in a feasible solution for the constrained version. Our algorithm repeatedly solves the LCP on sets of lanes which are candidates for coalitions. It's worth mentioning that time efficiency of LCP is another reason that we opt for using this method in our algorithm. If the solution of the base LCP satisfies all of the constraints, then we will have identified a feasible coalition. Otherwise, we modify the candidate coalition and repeat the process. We also modify feasible candidate coalitions to reduce the total system-wide cost. When the stopping conditions are met, we will have found a core cost allocation for each coalition selected. We use the following integer program to solve the base LCP on each candidate coalition $S \subseteq L$.

$$\text{LCPIP:} \quad \min \quad \sum_{i \in L} \sum_{j \in L} \rho_{head(i),tail(j)} \; f_{head(i),tail(j)} \; \alpha_{ij} \tag{4.1}$$

$$\text{s.t.} \quad \sum_{i \in L} \alpha_{ij} = 1 \qquad \forall j \in L \tag{4.2}$$

$$\sum_{j \in L} \alpha_{ij} = 1 \qquad \forall i \in L \tag{4.3}$$

$$\alpha_{ij} \in \{0,1\} \qquad \forall c \in C \tag{4.4}$$

The LCPIP simply assigns the head of each lane to the tail of one of the lanes. This is in fact the well known assignment problem which has zero integrality gap. At each iteration of our algorithm, we solve the LCPIP, decompose the solution into simple cycles, and check the constraints imposed on the cycles for feasibility. Our solution algorithm consists of two main phases: constructing a feasible coalition structure, and improvement. The details of our algorithms are presented in the following subsections.

4.1 Constructing a Feasible Coalition Structure

In the first phase of our solution approach, we search for a partitioning of the lane set L into at most MK subsets such that one subset corresponds to the set of lanes left out of the coalitions and the remaining subsets form core stable coalitions. We do this in two stages. First, we identify a coalition structure which satisfies C1 - C5, but not necessarily C6. Then, we modify the coalitions to satisfy C6.

In real-world problems, the lanes of each shipper are allowed to be included in a prespecified number of coalitions (i.e. MSK_i). This number can be specified by the managers and policy makers of the collaboration. It is clear that by increasing the number of coalitions which firms can participate in, they would be more likely to cooperate with a wider range of other shippers which can yield a better rate for them. In our algorithm, we first start to construct a set of coalitions for the smallest possible value of MSK_i (i.e. msk_i=1) and we record the obtained solution as the best solution. Afterwards, we increase the value of the msk_i in each iteration by one. In case we have a better solution, we update the best solution found. Otherwise, we terminate our algorithm. In addition to the worse solution, this process can be terminated by the time which msk_i exceedes its prespecified upperbound MSK_i .

The construction procedure of the initial core stable coalitions is presented in Algorithm 1. We construct the coalitions sequentially. We first put all of the lanes without a coalition together as a candidate coalition. We then reduce the candidate coalition to satisfy size limits and core stability conditions. Afterwards, we do the same processes for the remaining lanes, as a new candidate coalition. However, the shippers which cooperate with MSP_i other shippers (constraint C1) in previously constructed coalitions cannot enter their lanes in the candidate coalition. Hence, we remove the lanes of those shippers from the candidate list. Then, we solve the LCPIP on the remaining candidate lanes and check for additional constraints on the cycles.

In each iteration, if the LCPIP solution is not feasible for PCLCP, we try to remove one of the lanes causing the infeasibility. In order to identify this lane, first of all we must identify the cycle(s) which violates the constraint C1 the most, in its own. Afterwards, we have to identify the shipper(s) which cause this violation, in the on hand cycle. Then, we try to collapse the identified cycle(s) by extracting one of the lanes of the on hand shipper(s). Note that after each time we extract a lane from the obtained solution, we need to solve the LCPIP again for the remaining candidate lanes. However, in some cases despite the fact that there is no infeasible cycle, the union of the cycles may make the coalition infeasible. In such cases, we remove one of the lanes of the shipper with the most violated maximum partners constraint C1, from the candidate collaboration. It is worth mentioning that in case of ties in each part of the algorithm we make a random selection.

Once we reduce the candidate coalition to a core stable coalition, we first remove the lanes of the single-lane cycles from the candidate coalition because they do not make any positive contribution to the coalition. Afterwards, we check for the maximum lanes per coalition limit C5 because the candidate coalition may exceed it. Note that removing a whole cycle does not disturb the core stability of a coalition, so we do not have to solve the LCPIP again. If necessary, we reduce the size the coalition by removing the lanes of cycles in the LCPIP solutions. We repeatedly remove the lanes of the cycle with the highest *repositioning* cost, which is referred to as the most *expensive* cycle in the algorithms, until the number of lanes in the coalition is less than or equal to MKL. Then, we add the candidate coalition at hand to the coalition structure. We stop if we have reached the maximum coalition limit MK or we cannot find another core stable coalition.

In some cases, the obtained solution is made from only single-lane cycles which do not have positive contribution to the total saving, as mentioned before. Even though the probability of such cases is quite low, we have to note that such cases can be interpreted as a stopping condition of our algorithm.

Algorithm 1 Procedure for constructing core stable coalitions.

```
set the best solution equal to a large value
set the msk_i equal to 0
repeat
    increase msk_i by one unit
    repeat
    form a candidate coalition from lanes without a coalition
    remove the lanes of shippers which cooperate with more than MSP_i other
    shippers from candidate coalition
        repeat
        solve LCPIP on candidate coalition and decompose solution into
        simple cycles
            if candidate coalition is core stable then
            remove the lanes of single-lane cycles from candidate coalition
                while Number of lanes in candidate coalition exceeds MKL do
                Remove lanes of most expensive cycle from candidate coalition
                end while
                if candidate coalition is not empty then
                    add candidate coalition to list of coalitions
                    update set of lanes without a coalition
                end if
            else
                remove an infeasibility causing lane from candidate coalition
            end if
        until candidate coalition is core stable or empty
    until no additional stable coalitions can be found
          OR MK coalitions are found
    run algorithm 2
    run algorithm 3 (do improvement)
    if the objective function value is less than best solution then
         record the obtained solution as the best solution found
    else
        break
    end if
until msk_i equals MSK_i
```

The set of coalitions at hand still may not satisfy pairwise size difference constraint C6. Thus, we iteratively modify the coalitions to get a set of coalitions which satisfy all of the constraints. The steps of our procedure are listed in Algorithm 2. For each pair of coalitions which do not satisfy pairwise size difference limit, we have two options: First, we can increase the smaller coalition's size by adding a lane without a coalition. Second, we can remove some of the larger coalition's lanes. Preferably, we try adding lanes to the smaller coalition before removing lanes from the larger coalition. At each iteration, we take the smallest and the largest coalition in the current solution. We first try to increase the size of the smaller coalition. We first form a candidate coalition from the lanes of the smaller coalition together with the lanes without a coalition. We then remove the lanes from the candidate coalition in search for a smaller but core stable coalition (in order to decrease the difference between the size of coalitions and also obtaining a cost allocation in core). However, if the new solution's objective value is greater than the former one at any point, we do not remove any more lanes from the candidate coalition because the objective will surely get worse in subsequent iterations. Hence, in such cases, we will start to remove the lanes from the larger coalition instead. Since, removing all the lanes of a cycle might be an aggressive approach, we try to remove as few lanes as possible to satisfy the constraint C6. Therefore, first we identify the cycle with the most repositioning cost ratio. Then, we identify the lane which causes the least negative effect (i.e. most positive impact) on the cycle's repositioning cost, in case we remove the lane on hand. After applying all these steps, we have to check whether the obtained solution satisfies the constraint C6 or not. Note that the objective value we are referring is the total system-wide cost including the costs of the coalitions and the lanes without a coalition.

Now, we have reached an initial solution which satisfies all the mentioned constraints. However, it does not necessarily offer the set of coalitions withe the least total repositioning cost. Hence, in upcoming section we will discuss some methods for improving the initial solution in detail.

Algorithm 2 Procedure for reducing pairwise size difference of coalitions

```
while maximum pairwise size difference of coalitions exceeds MKLD do
    L^1 ← smallest coalition in current solution
    L^2 ← largest coalition in current solution
    form a candidate coalition from union of L^1 and lanes without coalition
    while |L^2| − |L^1| > MKLD AND candidate coalition is larger than L^1 do
        solve LCPIP on candidate coalition
        if new objective is greater than old one then
            break while
        else
            if candidate coalition is core stable then
                remove lanes of single-lane cycles from candidate coalition
                while Number of lanes in candidate coalition exceeds MKL do
                    Remove lanes of most expensive cycle from candidate coalition
                end while
                if candidate coalition is larger than L^1 then
                    replace L^1 with candidate coalition
                    update set of lanes without a coalition
                end if
            else
                remove an infeasibility causing lane from candidate coalition
            end if
        end if
    end while
    while |L^2| − |L^1| > MKLD do
        Identify the cycle with the greatest amount of repositioning ratio in L^2
        remove a lane of identified cycle which has the least negative effect
                                (i.e. most positive impact) on the total cost
        update set of lanes without a coalition
    end while
end while
```

4.2 Coalition Improvement Algorithm

The coalition structure we have at the end of the procedure described in Algorithms 1 and 2 is not necessarily the lowest cost coalition structure for the given set of lanes. For this reason, we have devised an improvement algorithm with the aim of decreasing the total system-wide cost of the coalition structure.

According to theorem 2, as discussed before, the probability that all lanes form a core stable coalition is extremely low. However, in order to minimize the total system-wide gap, we try to exclude as few lanes from the grand coalition, as possible. Hence, by applying the improvement algorithm we are looking to lower the system-wide cost. In the improvement algorithm (see Algorithm 3), we try to uniformize the differences between actual partnership number and maximum partners limit of each shipper. In each major iteration, we first identify a shipper which is closest to its maximum partner limit for each coalition. Generally, these shippers deprive other shippers of a higher degree of collaboration. Adding more shippers to the coalition is likely to cause them to exceed their maximum partner limit which cause infeasibility, while the rest of the shippers in the coalition are willing to accept more collaborative partners. Then, we add the lanes of each of the aforementioned shippers to an exclusion list of the corresponding coalition. These lanes will be excluded from their current coalition in the search for a lower cost solution but they can be used in other coalitions. Next, we go through the set of coalitions in the solution. For each coalition L^k, we first form a candidate coalition. That consists of the union of the lanes of L^k except the lanes in its exclusion list, lanes without a coalition and the set of lanes in the exclusion lists of other coalitions. As before, we remove lanes from the candidate coalition in the hope of reaching a core stable subset. Once the remaining candidate coalition is deemed core stable, we check whether it includes any lanes from the exclusion lists of other coalitions and removing would disturb the core stability of other coalitions. If the core stability of the other coalitions remain intact, we replace L^k with the candidate coalition and update the coalitions and the exclusion lists as necessary. Once we go through all of the coalitions, we check whether there is any improvement in the objective. We update the coalitions in the case of a tangible improvement. Otherwise, we revert back to the previous set of coalitions. We stop when we cannot improve the current solution.

Now, after completing all steps, designed for constructing the set of coalitions, we increase the msk_i by one. In other words, we relax the constraint C4 and we let shippers to enter their lanes to more coalitions. Then, we repeat all of these steps for new msk_i until facing with either a worse solution (compared with former msk_i) or reaching the predetermined value of msk_i.

Algorithm 3 Procedure for improving the total cost of the coalition structure.

```
repeat
    for each coaliton in coalition structure do
        add lanes of shipper closest to its max partner limit to the coalition's exclusion list
    end for
    for each coaliton L^k in coalition structure do
        form a candidate coalition from union of lanes without a coalition
                                        and L^k minus its exclusion list
                                        and lanes in exclusion lists of other coalitions
        repeat
            solve LCPIP on candidate coalition
            decompose on hand solution into simple cycles
            if candidate coalition is core stable then
                remove lanes of single-lane cycles from candidate coalition
                while Number of lanes in candidate coalition exceeds MKL do
                    Remove lanes of most expensive cycle from candidate coalition
                end while
                if candidate coalition is not empty then
                    if candidate coalition
                      does not make other coalitions unstable then
                        replace L^k with candidate coalition
                        update exclusion list of L^k
                        remove lanes of new L^k from their exclusion lists
                                                        and other coalitions
                        update set of lanes without a coalition
                    else
                        empty candidate coalition
                    end if
                end if
            else
                remove an infeasibility causing lane from candidate coalition
            end if
        until candidate coalition is core stable or empty
    end for
    if new objective is not better than the previous one then
        revert coalition structure to the previous one
    end if
until no change in objective
```

4.3 Cost Allocation

Once the construction of the core stable coalitions procedure terminates, we determine the allocated cost for each lane included in one of the selected coalitions. Recall that we build the coalitions such that each has a non-empty cost allocation core. In the cost allocation phase, we search for a cost allocation in the core. That is, we look for a cost allocation that is budget-balanced and stable (or group strategy proof). Since there can be multiple cost allocations in the core, we search for the one which makes the percentage savings of the lanes as close to each other as possible.

Our cost allocation procedure is based on solving a linear program with a huge number of rows, one for each feasible cycle (see (3.3)). Therefore, we formulate a cost allocation master problem with a limited number of rows and add more rows corresponding to the violated constraints as needed by solving a row generation sub-problem.

Let L^k denote the set of lanes in coalition k. Let C^k denote the set of all feasible cycles consisting only of the lanes of L^k. Let $\hat{C}^k$ denote the set of cycles given by the LCPIP solution on L^k. Let w_l be the decision variable corresponding to the cost allocated to lane $l \in L^k$. Let R_{min} and R_{max} be the decision variables corresponding to the minimum and maximum savings of the lanes in the coalition, respectively. Our cost allocation linear problem (CALP) is given by (4.5)-(4.12).

$$
\begin{aligned}
\text{CALP:} \quad & \min \quad R_{max} - R_{min} & & (4.5)\\
& \text{s.t.} \quad w_l \leq g_l R_{max} & \forall l \in L^k \quad & (4.6)\\
& \quad w_l \geq g_l R_{min} & \forall l \in L^k \quad & (4.7)\\
& \quad \sum_{l \in L_c} w_l = f_c & \forall c \in \hat{C}^k \quad & (4.8)\\
& \quad \sum_{l \in L_c} w_l \leq f_c & \forall c \in C^k - \hat{C}^k \quad & (4.9)\\
& \quad w_l \geq 0 & \forall l \in L^k \quad & (4.10)\\
& \quad R_{max} \geq 0 & & (4.11)\\
& \quad R_{min} \geq 0 & & (4.12)
\end{aligned}
$$

In the CALP formulation, the objective (4.5) minimizes the difference between the maximum and minimum percentage savings of the lanes in the coalition. Constraints (4.6) and (4.7) assures that R_{max} and R_{min} take appropriate values. Constraints (4.8) and (4.9) are the budget balance and stability constraints, respectively. Note that (3.1) and (3.3) imply (4.8). The rest are non-negativity constraints.

Since the CALP formulation contains a potentially huge number of stability constraints (4.9), we first relax the CALP by removing the stability constraints, and we only add the stability constraints violated by the relaxation to the model. We will refer to this relaxed model the restricted cost allocation master problem (RCAMP). Let $\tilde{C}$ be the set of cyles corresponding to the stability constraints included in the RCAMP. We start by solving the RCAMP with an empty $\tilde{C}$. We then look for a violated stability constraint not included in the RCAMP. If we identify a violated stability constraint, we add it to the RCAMP and solve it again. If we cannot find any violated stability constraints, it means that we have found a cost allocation in the core, and we stop.

We check the existence of a violated stability constraint by solving the row generation subproblem (RGSP) given by (4.13) - (4.25), where $\tilde{w}_l, l \in L^k$ are the current costs allocated by the RCAMP.

$$
\begin{aligned}
\text{RGSP:}\ \min \quad & \sum_{l\in L}(f_l-\tilde{w}_l)r_l+\sum_{a\in A}\rho_a f_a t_a && && (4.13)\\
\text{s.t.} \quad & \sum_{(m,n)\in L} r_{(m,n)}+\sum_{(m,n)\in A} t_{(m,n)} && && \\
& -\sum_{(n,m)\in L} r_{(n,m)}-\sum_{(n,m)\in A} t_{(n,m)}=0 && \forall n\in N && (4.14)\\
& \sum_{(m,n)\in L} r_{(m,n)}+\sum_{(m,n)\in A} t_{(m,n)}\le 1 && \forall n\in N && (4.15)\\
& \sum_{j\in P\setminus\{i\}} z_{ij}\le MSP_i && \forall i\in P && (4.16)\\
& \sum_{l\in L_i} r_l-|L_i|q_i\le 0 && \forall i\in P && (4.17)\\
& \sum_{l\in L_i} r_l-q_i\ge 0 && \forall i\in P && (4.18)\\
& q_i-z_{ij}\ge 0 && \forall\{i,j\}\subseteq P && (4.19)\\
& q_j-z_{ij}\ge 0 && \forall\{i,j\}\subseteq P && (4.20)\\
& q_i+q_j-z_{ij}\le 1 && \forall\{i,j\}\subseteq P && (4.21)\\
& r_l\in\{0,1\} && \forall l\in L^k && (4.22)\\
& t_a\in\{0,1\} && \forall a\in A && (4.23)\\
& q_i\in\{0,1\} && \forall i\in P && (4.24)\\
& z_{ij}\in\{0,1\} && \forall\{i,j\}\subseteq P && (4.25)
\end{aligned}
$$

In the RGSP, constraint (4.14) satisfies the flow balance for each node. In other words, it makes the number of arcs (loaded plus empty truck movements) entering and leaving each node equal to each other. Constraints (4.15) ensure that any feasible solution only consists of simple cycles. Constraints (4.16) enforce partner bounds C1. Constraints (4.17) and (4.18) force the value of variable q_i to 1, when any lane of shipper i is selected; and to take the value of 0, when none of the lanes of the shipper is selected. Constraints (4.19)-(4.21) ensure that decision variable z_{ij} equals 1 if and only if lanes of both shipper i and shipper j are selected.

It is important to note that the solution of the RGSP may include more than one simple cycle. Therefore, the formulation presented above is actually a relaxation of the true row generation subproblem. Its objective value cannot be lower than the negative of the highest stability violation. We could have added additional constraints to the formulation to prevent multiple cycles in the solution. However, these constraints would have a negative impact on the tractability of the RGSP and, more importantly, they are not really necessary. If the RGSP objective is negative, then all of the cycles in the solution must have nonnegative stability violation. Otherwise, we can remove the cycle without the violation and obtain a lower objective value, which contradicts with the optimality of the solution. When the RGSP has a negative objective, we add the stability constraints of all of the cycles with positive stability violation to the RCAMP.

5. COMPUTATIONAL EXPERIMENTS

We have assessed the performance of our algorithms on randomly generated Euclidean problem instances. We have generated these instances by modifying the instance generation procedure of Ergun et al. [10] so that each lane belongs to one of a given number of shippers and each shipper has a partner limit. Each instance contains nodes which were randomly dispersed over a square region of $1,800 \times 1,800$ miles. Some of the nodes are placed in clusters corresponding concentration of points as in metropolitan areas. Each generated node has at least one incoming or outgoing lanes. Some of the nodes have incoming lanes only, some have outgoing lanes only, while some have both incoming and outgoing lanes. Lanes are prevented from originating and terminating in the same cluster.

It worth mentioning that there are two main types of instances: (i) large shipper instances in which there are $|L|/20$ shippers with 15 to 25 lanes and partner limit (MSP_i) of 1 or 2, and (ii) small shipper instances in which there are $|L|/10$ shippers with 5 to 15 lanes, and partner limit of 3 or 4.

All of the algorithms were implemented using the Java programming language. The experiments were run on a desktop PC with 3.60 GHz Intel i7-4790 processor, 16 GB RAM, and Microsoft Windows 8 Professional operating system. IBM ILOG CPLEX 12.6 and Concert technology were utilized to solve the linear and integer programs.

In the experiments, we set MKL and $MKLD$ to 60% and 20% of the number of lanes of the instance, respectively. We varied MK and MSK_i to analyze their impact on the final solutions. We evaluated the performance of our solution approach in terms of the numbers of coalitions formed, the number of lanes in each coalition, the total number of lanes in all of the coalitions formed, the allocated cost per mile for each lane, the optimality gap of the total system-wide cost with respect to the LCP lower bound, and the CPU times.

5.1 Results

In this section, we first present the results of experiments with the small shipper instances and then we present the experiments with large shipper instances.

For each set of instances, we first discuss the results of our experiments in which we set $MSK_i = 1$ and vary the maximum number of coalitions MK. The results are summarized in Tables 5.1 and 5.3 for small shippers and also 5.2 and 5.4 for large shippers. Presented results in Tables 5.1 and 5.2 correspond to the size of coalitions and percentage of the lanes which are participated in the coalitions. Upon inspecting Table 5.1 and 5.2, we observe that as MK increases, the number of and percentage of lanes placed in a coalition increases. The size of the largest and the smallest coalition decreases as well. Because we construct the initial set of coalitions sequentially, later coalitions have fewer lanes to choose from than earlier coalitions.

Table 5.3 and 5.4 lists statistics on the per mile cost allocated to the lanes selected for the coalitions. We report the maximum, minimum and average values of the mean and the standard deviation of cost per mile across different coalitions. Note that a low mean cost per mile for a coalition is an indication that the lanes of the coalition can be covered with low repositioning, and vice versa. We observe that increasing MK may result in an increased or decreased average mean and standard deviation of cost per mile.

Table 5.1: Number of lanes in coalitions when $MSK_i = 1$ (Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MK	Coal. sizes			Coals.	Lanes with Coal.	
					Max	Min	Avg		Count	%
1	100	100	10	1	40	40	40.0	1	40	40.0
				2	40	40	40.0	2	80	80.0
				3	29	9	22.3	3	67	67.0
				UNR	29	9	22.3	3	67	67.0
2	100	100	10	1	48	48	48.0	1	48	48.0
				2	44	31	37.5	2	75	75.0
				3	33	14	26.0	3	78	78.0
				UNR	33	14	26.0	3	78	78.0
3	100	200	20	1	30	30	30.0	1	30	15.0
				2	31	30	30.5	2	61	30.5
				3	31	30	30.7	3	92	46.0
				UNR	31	30	30.7	3	92	46.0
4	100	200	20	1	30	30	30.0	1	30	15.0
				2	37	28	32.5	2	65	32.5
				3	39	20	32.3	3	97	48.5
				UNR	39	20	34.0	4	136	68.0
5	200	200	20	1	41	41	41.0	1	41	20.5
				2	55	32	43.5	2	87	43.5
				3	41	34	37.3	3	112	56.0
				UNR	40	34	36.7	4	147	73.5
6	200	200	20	1	45	45	45.0	1	45	22.5
				2	45	41	43.0	2	86	43.0
				3	45	31	39.0	3	117	58.5
				UNR	45	6	29.8	5	149	74.5
7	200	400	40	1	40	40	40.0	1	40	10.0
				2	47	40	40.5	2	90	22,5
				3	47	40	44.7	3	134	33.5
				UNR	56	13	32.2	9	290	72.5
8	200	400	40	1	37	37	37.0	1	37	9.3
				2	37	28	32.5	2	65	16.3
				3	37	27	30.7	3	92	23.0
				UNR	42	24	31.2	8	250	62.5

Table 5.2: Number of lanes in coalitions when $MSK_i = 1$ (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MK	Coal. sizes			Coals.	Lanes with Coal.	
					Max	Min	Avg		Count	%
1	100	100	5	1	42	42	42.0	1	42	42.0
				2	44	41	42.5	2	85	85.0
				3	44	41	42.5	2	85	85.0
				UNR	44	41	42.5	2	85	85.0
2	100	100	5	1	56	56	56.0	1	56	56.0
				2	57	42	49.5	2	99	99.0
				3	57	42	49.5	2	99	99.0
				UNR	57	42	49.5	2	99	99.0
3	100	200	10	1	57	57	57.0	1	57	28.5
				2	53	51	52.0	2	104	52.0
				3	56	6	48.7	3	146	73.0
				UNR	50	29	42.0	4	168	84.0
4	100	200	10	1	47	47	47.0	1	47	23.5
				2	41	39	40.0	2	80	40.0
				3	41	48	39.3	3	118	59.0
				UNR	41	19	35.0	5	175	87.5
5	200	200	10	1	53	53	53.0	1	53	26.5
				2	54	50	52.0	2	104	52.0
				3	54	43	49.0	3	147	73.5
				UNR	49	36	41.5	4	166	83.0
6	200	200	10	1	58	58	58.0	1	58	29.0
				2	58	40	49.0	2	98	49.0
				3	58	39	46.0	3	138	69.0
				UNR	58	19	36.0	5	180	90.0
7	200	400	20	1	56	56	56.0	1	56	14.0
				2	56	53	54.5	2	109	27.3
				3	56	49	52.7	3	158	39.5
				UNR	56	42	42.1	8	337	84.3
8	200	400	20	1	59	59	59.0	1	59	14.8
				2	58	42	50.0	2	100	25.0
				3	60	33	49.7	3	149	37.2
				UNR	66	18	41.3	8	330	82.5

Table 5.3: Allocated cost per mile (w_l/f_l) when $MSK_i = 1$ ($\forall i \in P$)(Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MK	Mean			Std Dev		
					Max	Min	Avg	Max	Min	Avg
1	100	100	10	1	1.27	1.27	1.27	0.44	0.44	0.44
				2	1.29	1.14	1.22	0.55	0.45	0.50
				3	1.22	1.10	1.17	0.57	0.07	0.38
				UNR	1.22	1.10	1.17	0.57	0.07	0.38
2	100	100	10	1	1.11	1.11	1.11	0.39	0.39	0.39
				2	1.14	1.13	1.13	0.40	0.39	0.40
				3	1.37	1.05	1.19	0.47	0.39	0.42
				UNR	1.37	1.05	1.19	0.47	0.39	0.42
3	100	200	20	1	1.23	1.23	1.23	0.37	0.37	0.37
				2	1.34	1.23	1.28	0.37	0.31	0.34
				3	1.34	1.23	1.28	0.48	0.31	0.39
				UNR	1.34	1.23	1.28	0.48	0.31	0.39
4	100	200	20	1	1.20	1.20	1.20	0.41	0.41	0.41
				2	1.31	1.18	1.25	0.39	0.37	0.38
				3	1.31	1.21	1.27	0.52	0.41	0.48
				UNR	1.31	1.21	1.26	0.52	0.41	0.49
5	200	200	20	1	1.18	1.18	1.18	0.27	0.27	0.27
				2	1.35	1.14	1.25	0.60	0.30	0.33
				3	1.22	1.18	1.19	0.37	0.27	0.31
				UNR	1.23	1.17	1.20	0.67	0.22	0.38
6	200	200	20	1	1.20	1.20	1.20	0.37	0.37	0.37
				2	1.20	1.18	1.19	0.45	0.37	0.41
				3	1.27	1.18	1.22	0.46	0.37	0.43
				UNR	1.74	1.17	1.20	0.35	0.07	0.28
7	200	400	40	1	1.28	1.28	1.28	0.29	0.29	0.29
				2	1.28	1.20	1.24	0.43	0.29	0.36
				3	1.35	1.20	1.27	0.43	0.29	0.35
				UNR	1.35	1.20	1.27	0.42	0.24	0.35
8	200	400	40	1	1.21	1.21	1.21	0.33	0.33	0.33
				2	1.21	1.14	1.18	0.37	0.33	0.35
				3	1.21	1.14	1.17	0.37	0.26	0.32
				UNR	1.30	1.14	1.21	0.60	0.27	0.39

Table 5.4: Allocated cost per mile (w_l/f_l) when $MSK_i = 1$ ($\forall i \in P$) (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MK	Mean			Std Dev		
					Max	Min	Avg	Max	Min	Avg
1	100	100	5	1	1.27	1.27	1.27	0.37	0.37	0.37
				2	1.25	1.23	1.24	0.46	0.36	0.41
				3	1.25	1.23	1.24	0.46	0.36	0.41
				UNR	1.25	1.23	1.24	0.46	0.36	0.41
2	100	100	5	1	1.18	1.18	1.18	0.47	0.47	0.47
				2	1.19	1.17	1.18	0.57	0.44	0.51
				3	1.19	1.17	1.18	0.57	0.44	0.51
				UNR	1.19	1.17	1.18	0.57	0.44	0.51
3	100	200	10	1	1.26	1.26	1.26	0.31	0.31	0.31
				2	1.25	1.20	1.22	0.46	0.45	46.00
				3	1.28	1.22	1.25	0.42	0.22	0.31
				UNR	1.34	1.23	1.27	0.41	0.32	0.38
4	100	200	10	1	1.20	1.20	1.20	0.36	0.36	0.36
				2	1.23	1.23	1.23	0.49	0.42	0.46
				3	1.20	1.21	1.22	0.57	0.42	0.48
				UNR	1.28	1.21	1.24	0.52	0.35	0.43
5	200	200	10	1	1.15	1.15	1.15	0.40	0.40	0.40
				2	1.25	1.24	1.25	0.52	0.34	0.43
				3	1.25	1.24	1.25	0.52	0.34	0.42
				UNR	1.36	1.16	1.27	0.40	0.32	0.36
6	200	200	10	1	1.13	1.13	1.13	0.32	0.32	0.32
				2	1.19	1.13	1.16	0.38	0.32	0.35
				3	1.21	1.17	1.19	0.46	0.42	0.45
				UNR	1.34	1.13	1.22	0.48	0.32	0.42
7	200	400	20	1	1.22	1.22	1.22	0.39	0.39	0.39
				2	1.28	1.22	1.25	0.39	0.39	0.39
				3	1.28	1.21	1.24	0.32	0.39	0.40
				UNR	1.42	1.21	1.27	0.42	0.34	0.37
8	200	400	20	1	1.17	1.17	1.17	0.38	0.38	0.38
				2	1.24	1.23	1.23	0.38	0.37	0.37
				3	1.23	1.16	1.21	0.40	0.26	0.32
				UNR	1.28	1.17	1.23	0.49	0.22	0.70

Next, we discuss the impact of changing MSK_i on the results of the algorithm. In this group of experiments, we set MSK_i to one, two, or unbounded. We set MK to three when MSK_i is one or two. However, we set MK to unbounded when MSK_i is unbounded in order to monitor the algorithm's performance with the most possible relaxed form. Tables 5.5 - 5.8 present statistics collected on lanes with coalitions and allocated cost per mile, respectively. We expect the algorithm to construct as many coalitions as possible in order to get a better solution. However, in some cases we observe that despite the increase in the MSK_i, the number of selected coalitions does not change. This observation clearly indicates that increasing MSK_i does not always have a favorable impact on our solution because increasing MSK_i also expands the number of possible alternatives. In other words, Increasing the value of MSK_i and/or the number of lanes increases the number of possible cycles which can be used in the LCP solution. Because of tight partner bounds MSP_i, the LCP solution will be more likely to be infeasible for the PCLCP. Hence, we have to be more careful in selecting which lane to remove next from the candidate coalition. Therefore, our algorithm would struggle in removing the lanes in the right sequence for loose problem instances and bounds. That is why our algorithm prefers to keep the former solution, obtained with a smaller MSK_i, which has a smaller optimality gap.

Table 5.7 and 5.8 lists statistics similar to Table 5.3 and 5.4, but with varying MSK_i values, on allocated cost per mile for lanes in the selected coalitions. Due to the mentioned reasons in the previous paragraph, increasing MSK_i does not cause any change in the average and the standard deviation (because the solution is similar to the former one) in some instances, but in the rest of them, it leads to increased average mean cost per mile, while leading to decreased average standard deviation and increased maximum mean cost per mile. The negative correlation between MSK_i and allocated cost per mile is quite counterintuitive. We are essentially relaxing the problem by increasing MSK_i. So, we expect to see a decrease in repositioning costs incurred, but we observe the opposite. This is due to the fact that when MK is equal to two or higher, we must pay attention to the pairwise size difference of the coalitions to maintain fairness across the coalitions. Hence, there is a higher chance that the lanes will be placed in coalitions of higher repositioning.

Table 5.5: Number of lanes in coalitions when MSK_i varies (Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	Coal. sizes			Coals.	Lanes with Coal.	
					Max	Min	Avg		Count	%
1	100	100	10	1	29	9	22.3	3	67	67.0
				2	37	17	30.0	3	90	90.0
				UNR	29	9	22.3	3	67	67.0
2	100	100	10	1	33	14	26.0	3	78	78.0
				2	33	14	26.0	3	78	78.0
				UNR	25	5	15.7	6	94	94.0
3	100	200	20	1	31	30	30.7	3	92	46.0
				2	31	30	30.7	3	92	46.0
				UNR	27	2	11.0	16	176	88.0
4	100	200	20	1	39	20	32.3	3	97	48.5
				2	39	20	32.3	3	97	48.5
				UNR	33	4	18.4	9	166	83.0
5	200	200	20	1	41	34	37.3	3	112	56.0
				2	41	34	37.3	3	112	56.0
				UNR	46	6	18.2	9	164	82.0
6	200	200	20	1	45	31	39.0	3	117	58.5
				2	45	31	39.0	3	117	58.5
				UNR	45	6	29.8	5	149	74.5
7	200	400	400	1	47	40	44.7	3	134	33.5
				2	47	40	44.7	3	134	33.5
				UNR	56	13	32.2	9	290	72.5
8	200	400	400	1	37	27	30.7	3	92	23.0
				2	37	27	30.7	3	92	23.0
				UNR	31	2	15.0	20	299	74.8

Note: $MK = 3$ for $MSK_i \in \{1,2\}$; $MK = \infty$ for $MSK_i = \infty$.

Table 5.6: Number of lanes in coalitions when MSK_i varies (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	Coal. sizes			Coals.	Lanes with Coal.	
					Max	Min	Avg		Count	%
1	100	100	5	1	44	41	42.5	2	85	85.0
				2	41	22	33.0	3	99	99.0
				UNR	33	13	22.5	4	90	90.0
2	100	100	5	1	57	42	49.5	2	99	99.0
				2	57	42	49.5	2	99	99.0
				UNR	57	42	49.5	2	99	99.0
3	100	200	10	1	56	6	48.7	3	146	73.0
				2	56	36	48.7	3	146	73.0
				UNR	50	29	42.0	4	168	84.0
4	100	200	10	1	41	48	39.3	3	118	59.0
				2	41	8	39.3	3	118	59.0
				UNR	41	19	35.0	5	175	87.5
5	200	200	10	1	54	43	49.0	3	147	73.5
				2	54	43	49.0	2	98	49.0
				UNR	49	36	41.5	4	166	83.0
6	200	200	10	1	58	39	46.0	3	138	69.0
				2	58	39	46.0	3	138	69.0
				UNR	58	19	36.0	5	180	90.0
7	200	400	20	1	56	49	52.7	3	158	39.5
				2	56	49	52.7	3	158	39.5
				UNR	56	22	42.1	8	337	84.3
8	200	400	20	1	60	33	49.7	3	149	37.2
				2	56	49	52.7	3	158	39.5
				UNR	66	18	41.3	8	330	82.5

Note: $MK = 3$ for $MSK_i \in \{1,2\}$; $MK = \infty$ for $MSK_i = \infty$.

Table 5.7: Allocated cost per mile as MSK_i varies (Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	Mean			Std Dev		
					Max	Min	Avg	Max	Min	Avg
1	100	100	10	1	1.22	1.10	1.17	0.57	0.07	0.38
				2	1.13	1.18	1.23	0.47	0.43	0.45
				UNR	1.25	1.10	1.17	0.57	0.07	0.38
2	100	100	10	1	1.37	1.05	1.19	0.47	0.39	0.42
				2	1.37	1.05	1.19	0.47	0.39	0.42
				UNR	1.31	1.04	1.20	0.54	0.25	0.36
3	100	200	20	1	1.34	1.23	1.28	0.48	0.31	0.39
				2	1.34	1.23	1.28	0.48	0.31	0.39
				UNR	1.72	1.19	1.43	0.48	0.22	0.27
4	200	200	20	1	1.31	1.21	1.27	0.52	0.41	0.48
				2	1.31	1.21	1.27	0.52	0.41	0.48
				UNR	1.78	1.16	1.34	0.49	0.00	0.36
5	200	200	20	1	1.22	1.18	1.19	0.37	0.27	0.31
				2	1.22	1.18	1.19	0.37	0.27	0.31
				UNR	1.42	1.14	1.28	0.49	0.19	0.36
6	200	200	20	1	1.27	1.18	1.22	0.46	0.37	0.43
				2	1.27	1.18	1.22	0.46	0.37	0.43
				UNR	1.74	1.17	1.20	0.35	0.07	0.28
7	200	400	40	1	1.35	1.20	1.27	0.43	0.29	0.35
				2	1.35	1.20	1.27	0.43	0.29	0.35
				UNR	1.35	1.20	1.27	0.42	0.24	0.35
8	200	400	40	1	1.21	1.14	1.17	0.37	0.26	0.32
				2	1.21	1.14	1.17	0.37	0.27	0.32
				UNR	1.49	1.14	1.33	0.50	0.00	0.33

Note: $MK = 3$ for $MSK_i \in \{1,2\}$; $MK = \infty$ for $MSK_i = \infty$.

Table 5.8: Allocated cost per mile as MSK_i varies (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	Mean			Std Dev		
					Max	Min	Avg	Max	Min	Avg
1	100	100	5	1	1.25	1.23	1.24	0.46	0.36	0.41
				2	1.27	1.22	1.25	0.43	0.33	0.37
				UNR	1.38	1.22	1.28	0.37	0.29	0.23
2	100	100	5	1	1.19	1.17	1.18	0.57	0.44	0.51
				2	1.19	1.17	1.18	0.57	0.44	0.51
				UNR	1.19	1.17	1.18	0.57	0.44	0.51
3	100	200	10	1	1.28	1.22	1.25	0.42	0.22	0.31
				2	1.28	1.22	1.25	0.42	0.22	0.31
				UNR	1.34	1.23	1.27	0.41	0.32	0.38
4	100	200	10	1	1.20	1.21	1.22	0.57	0.42	0.48
				2	1.23	1.21	1.22	0.52	0.42	0.48
				UNR	1.28	1.21	1.24	0.52	0.35	0.43
5	200	200	10	1	1.25	1.24	1.25	0.52	0.34	0.42
				2	1.25	1.24	1.25	0.52	0.34	0.42
				UNR	1.36	1.16	1.27	0.39	0.32	0.36
6	200	200	10	1	1.21	1.17	1.19	0.46	0.42	0.45
				2	1.21	1.17	1.19	0.46	0.42	0.45
				UNR	1.34	1.13	1.22	0.48	0.32	0.42
7	200	400	20	1	1.28	1.21	1.24	0.32	0.39	0.40
				2	1.28	1.21	1.24	0.32	0.39	0.40
				UNR	1.42	1.21	1.27	0.42	0.34	0.37
8	200	400	20	1	1.23	1.16	1.21	0.40	0.26	0.32
				2	1.28	1.21	1.24	0.42	0.39	0.40
				UNR	1.28	1.17	1.23	0.49	0.23	0.37

Note: $MK = 3$ for $MSK_i \in \{1,2\}$; $MK = \infty$ for $MSK_i = \infty$.

Tables 5.9 and 5.10 list the optimality gaps of our final solutions with respect to the LCP lower bound for each instance. We assume that the lanes without a coalition will be covered by single lane cycles. We take the total cost of a coalition structure as the sum of the costs of the coalitions and the single lane cycles of the lanes without a coalition. Then, we compute the optimality gap by calculating the difference between the total cost of the coalition structure and LCP lower bound as a percentage of the LCP lower bound for each instance. Despite the fact that increasing MK tend to decrease the optimality gap, we observe that in most of the cases increasing MSK_i tends to increase the optimality gap.This fact also can be accounted by the discussed reasons in previous parts. In short, we can claim that, not only increasing the value of MSK_i is not an advantage for our algorithm, but also it makes the coalition construction processes more complicated and more challenging. Consequently, our algorithm prefers to keep the solution with the smaller MSK_i as best solution found. Since larger MSK_i tends to increase the optimality gap in practice, the algorithm would terminate and would use the best solution found so far for the larger MSK_i. These tables also show that larger instances are associated with larger optimality gaps.

Table 5.9: Optimality gaps with respect to the LCP lower bound (Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	MK 1	2	3	∞
1	100	100	10	1	25%	10%	9%	9%
				2	*	10%	4%	9%
				Unr	*	*	4%	9%
2	100	100	10	1	30%	15%	14%	14%
				2	*	15%	14%	7%
				Unr	*	*	14%	7%
3	100	200	20	1	42%	34%	27%	27%
				2	*	34%	27%	21%
				Unr	*	*	27%	14%
4	100	200	20	1	49%	42%	34%	24%
				2	*	42%	34%	18%
				Unr	*	*	34%	18%
5	200	200	20	1	45%	33%	28%	16%
				2	*	33%	28%	16%
				Unr	*	*	28%	16%
6	200	200	20	1	46%	35%	28%	18%
				2	*	35%	28%	18%
				Unr	*	*	27%	18%
7	200	400	40	1	50%	43%	38%	19%
				2	*	43%	38%	19%
				Unr	*	*	38%	19%
8	200	400	40	1	55%	51%	48%	27%
				2	*	51%	48%	25%
				Unr	*	*	48%	25%

*: Same as previous line.

Table 5.11 and 5.12 contains the number of coalitions selected under different combinations of MSK_i and MK values. We observe that the number of selected coalitions is mainly determined by MK, as expected. On the other hand, it is affected by both MSK_i and MK, especially for small MSK_i. When MSK_i is small, the number of selected coalitions is highly likely to be small, even with unbounded MK. The number of coalitions is not always high even when both MSK_i and MK are unbounded.

As mentioned before, the small shipper instances have looser partnership constraints, compared to the large shippers. Hence, we might expect better solution at first glance. However, by observing the presented solutions, we find out that the solutions of the large instances are comparatively more favorable. After a detailed scrutiny of these two instance types, we maintain that the range difference between the lane number of each shipper can be an account for this fact. Even though small shippers have looser partnership constraints, they will lack the opportunity to have a higher degree of collaboration, due to the small number of lanes they have. On the other hand, larger shippers have more freedom to participate in coalitions without violating the partnership constraints.

Table 5.10: Optimality gaps with respect to the LCP lower bound (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	MK 1	2	3	∞
1	100	100	5	1	30%	10%	10%	10%
				2	*	10%	3%	6%
				Unr	*	*	3%	6%
2	100	100	5	1	24%	2%	2%	2%
				2	*	2%	2%	2%
				Unr	*	*	2%	2%
3	100	200	10	1	35%	22%	12%	9%
				2	*	22%	12%	9%
				Unr	*	*	12%	9%
4	100	200	10	1	37%	30%	21%	7%
				2	*	30%	21%	7%
				Unr	*	*	21%	7%
5	200	200	10	1	35%	26%	15%	12%
				2	*	26%	15%	12%
				Unr	*	*	15%	12%
6	200	200	10	1	42%	30%	21%	9%
				2	*	30%	21%	9%
				Unr	*	*	21%	9%
7	200	400	20	1	42%	36%	30%	10%
				2	*	36%	30%	10%
				Unr	*	*	30%	10%
8	200	400	20	1	52%	47%	39%	17%
				2	*	47%	39%	17%
				Unr	*	*	39%	17%

*: Same as previous line.

Tables 5.13 and 5.14 contains the CPU times of the algorithm for different MSK_i and MK combinations on the problem instances. The CPU time spent on finding coalitions is more than finding a cost allocation in most cases. We observe our heuristic finds high quality solutions for tight problems very quickly. As the problem becomes looser, the CPU time increases and the solution quality degrades. For $MSK_i = 1$, the algorithm terminates in under a minute with the small instances, and well under one hour for the largest instances. The algorithm does not take more than two hours in any of the experiments.

Table 5.11: Number of coalitions selected(Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	MK 1	2	3	∞
1	100	100	10	1	1	2	3	3
				2	*	2	3	3
				Unr	*	*	3	3
2	100	100	10	1	1	2	3	3
				2	*	2	3	6
				Unr	*	*	3	6
3	100	200	20	1	1	2	3	3
				2	*	2	3	10
				Unr	*	*	3	16
4	100	200	20	1	1	2	3	3
				2	*	2	3	9
				Unr	*	*	3	9
5	200	200	20	1	1	2	3	4
				2	*	2	3	9
				Unr	*	*	3	9
6	200	200	20	1	1	2	3	5
				2	*	2	3	5
				Unr	*	*	3	5
7	200	400	40	1	1	2	3	9
				2	*	2	3	9
				Unr	*	*	3	9
8	200	400	40	1	1	2	3	8
				2	*	2	3	20
				Unr	*	*	3	25

*: Same as previous line.

Table 5.12: Number of coalitions selected (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	MK			
					1	2	3	∞
1	100	100	5	1	1	2	2	2
				2	*	2	3	4
				Unr	*	*	3	4
2	100	100	5	1	1	2	2	2
				2	*	2	2	2
				Unr	*	*	2	2
3	100	200	10	1	1	2	3	4
				2	*	2	3	4
				Unr	*	*	3	4
4	100	200	10	1	1	2	3	5
				2	*	2	3	5
				Unr	*	*	3	5
5	200	200	10	1	1	2	3	4
				2	*	2	3	4
				Unr	*	*	3	4
6	200	200	10	1	1	2	3	5
				2	*	2	3	5
				Unr	*	*	3	5
7	200	400	20	1	1	2	3	8
				2	*	2	3	8
				Unr	*	*	3	8
8	200	400	20	1	1	2	3	8
				2	*	2	3	8
				Unr	*	*	3	8

*: Same as previous line.

Table 5.13: CPU times (Small Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	$MK=1$			$MK=2$			$MK=3$			$MK=\infty$		
					t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}
1	100	100	10	1	7.00	19.31	26.31	4.30	3.51	7.81	7.49	13.02	20.51	7.37	13.00	20.37
				2	*	*	*	9.25	35.06	44.31	16.60	37.59	54.19	14.17	14.31	28.48
				Unr	*	*	*	*	*	*	32.88	37.54	70.42	14.09	13.11	27.20
2	100	100	10	1	8.95	16.23	25.18	4.88	21.41	26.29	6.07	20.50	26.57	6.06	26.65	32.71
				2	*	*	*	10.09	20.74	30.83	13.71	20.42	34.13	18.24	13.25	31.49
				Unr	*	*	*	*	*	*	13.83	20.45	34.28	29.81	12.78	42.59
3	100	200	20	1	49.59	16.85	66.44	52.86	27.50	80.36	44.33	36.25	80.58	44.43	36.98	81.41
				2	*	*	*	129.10	24.14	153.24	185.17	39.19	224.36	189.25	33.06	222.31
				Unr	*	*	*	*	*	*	38.06	183.82	221.88	703.35	36.52	739.87
4	100	200	20	1	70.18	9.51	79.69	73.90	33.41	107.31	52.09	35.98	88.07	140.14	41.67	181.81
				2	*	*	*	182.12	30.68	212.80	135.14	35.91	171.05	139.83	41.16	180.99
				Unr	*	*	*	*	*	*	134.74	34.04	168.78	358.12	43.53	401.65
5	200	200	20	1	134.34	177.70	312.04	71.23	492.84	564.07	47.82	401.17	448.99	43.37	530.08	573.45
				2	*	*	*	136.03	482.27	618.30	185.11	415.05	600.16	127.54	292.95	420.49
				Unr	*	*	*	*	*	*	182.71	412.79	595.50	254.01	304.42	558.43
6	200	200	20	1	42.28	235.28	277.56	53.92	422.92	476.84	51.41	488.18	539.59	66.39	602.81	669.20
				2	*	*	*	115.39	423.60	538.99	140.47	499.25	639.72	160.20	600.00	760.20
				Unr	*	*	*	*	*	*	136.98	485.52	622.50	159.14	592.90	752.04
7	200	400	40	1	886.85	157.20	1044.05	1421.42	391.30	1812.72	1623.70	645.20	2268.90	1789.80	1281.84	3071.64
				2	*	*	*	3090.92	389.80	3480.72	3757.77	647.25	4405.02	4817.74	1314.96	6132.70
				Unr	*	*	*	*	*	*	3789.93	639.30	4429.23	4830.71	1289.56	6120.27
8	200	400	40	1	857.40	115.38	972.78	1385.21	201.66	1586.87	1630.44	255.68	1886.12	1529.45	872.36	2401.81
				2	*	*	*	2909.56	236.45	3146.01	3715.10	256.05	3971.15	4972.93	828.53	5801.46
				Unr	*	*	*	*	*	*	3725.71	258.10	3983.81	12785.80	823.42	13609.22

*: Same as previous line.

Table 5.14: CPU times (Large Shippers)

Inst	$\|N\|$	$\|L\|$	$\|P\|$	MSK_i	$MK=1$			$MK=2$			$MK=3$			$MK=\infty$		
					t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}	t_{CF}	t_{CA}	t_{total}
1	100	100	5	1	4.19	38.30	42.49	4.75	41.28	46.03	4.82	42.40	47.22	4.79	41.73	46.52
				2	*	*	*	9.44	41.98	51.42	15.07	47.37	62.44	11.07	33.78	44.85
				Unr	*	*	*	*	*	*	30.55	45.37	75.92	17.73	32.31	50.04
2	100	100	5	1	5.21	28.68	33.89	3.50	38.04	41.54	3.57	38.20	41.77	3.61	38.00	41.61
				2	*	*	*	9.82	38.66	48.48	6.96	38.40	45.36	11.42	40.52	51.94
				Unr	*	*	*	*	*	*	6.99	38.21	45.20	11.40	38.49	49.89
3	100	200	10	1	57.24	18.87	76.11	68.73	39.47	108.20	62.96	64.11	127.07	45.90	81.20	127.10
				2	*	*	*	131.72	36.95	168.67	1.28	1.22	1.25	148.33	83.21	231.54
				Unr	*	*	*	*	*	*	129.81	61.23	191.04	147.11	78.40	225.51
4	100	200	10	1	62.08	15.04	77.12	56.14	16.45	72.59	55.17	28.12	83.29	50.99	59.07	110.06
				2	*	*	*	123.96	16.73	140.69	161.16	21.20	182.36	138.70	54.04	192.74
				Unr	*	*	*	*	*	*	161.94	29.07	191.01	139.10	58.92	198.02
5	200	200	10	1	56.97	314.92	371.89	50.85	588.11	638.96	44.73	863.21	907.94	66.11	785.50	851.61
				2	*	*	*	163.10	58.24	221.34	115.87	812.67	928.54	222.39	799.10	1021.49
				Unr	*	*	*	*	*	*	117.14	831.02	948.16	222.33	790.18	1012.51
6	200	200	10	1	37.78	300.19	337.97	48.02	482.21	530.23	64.65	790.94	855.59	43.28	941.27	984.55
				2	*	*	*	112.55	466.21	578.76	187.63	821.63	1009.26	245.06	945.01	1190.07
				Unr	*	*	*	*	*	*	187.63	798.01	985.64	244.52	936.48	1181.00
7	200	400	20	1	901.21	254.32	1155.53	1324.22	486.90	1811.12	1600.65	735.11	2335.76	1596.51	1535.05	3131.56
				2	*	*	*	2768.90	629.09	3397.99	3550.87	743.87	4294.74	5552.18	1559.45	7111.63
				Unr	*	*	*	*	*	*	3550.37	707.70	4258.07	5605.06	1577.59	7182.65
8	200	400	20	1	1121.81	298.47	1420.28	1223.92	391.20	1615.12	1620.79	566.73	2187.52	1253.84	1631.00	2884.84
				2	*	*	*	2731.60	402.10	3133.70	3602.11	541.93	4144.04	5118.05	1621.50	6739.55
				Unr	*	*	*	*	*	*	3591.12	577.23	4168.35	5174.61	1623.37	6797.98

*: Same as previous line.

6. CONCLUSION

In this thesis, we study the design of coalition structures which consist of multiple disjoint core stable coalitions of shippers interested in collaborative truckload transportation procurement. Each coalition must have a minimum cost collaborative solution with an acceptable cost allocation. Due to the complexity of the task hand, we propose a heuristic to find good quality solutions to this problem.

Our heuristic appears to work best when the degree of freedom is limited due to either the number of lanes on hand or the restrictions put in place by the decision makers. We suggest using this algorithm when the lane density does not exceed $200/(1800\text{miles} \times 1800\text{miles})$ and at most one coalition per shipper is acceptable. Limiting each shipper to at most one coalition should in fact be preferable by most shippers since participating in multiple coalitions will require more coordination effort compared to participating in a single coalition. We also suggest allowing a high number of coalitions for the participants to get the most benefit.

Choosing the next lane to remove from the candidate coalition is critical for solving problem instances with a high number of lanes and loose restrictions. In our future work, we plan to investigate different criteria for choosing the next lane to remove from the candidate coalition for increasing the effectiveness of our solution approach.

REFERENCES

[1] **Agarwal and Ö. Ergun.,** Network design and allocation mechanisms for carrier alliances in liner shipping. Operations Research, 58(6):1726–1742, 2010. ISSN0030-364X.

[2] **R. Agarwal, Ö. Ergun, L., Houghtalen, and O. O. Ozener.,** Collaboration in cargo transportation. In Optimization and Logistics Challenges in the Enterprise, pages 373–409. 2009. ISBN 978-0-387-88616-9.

[3] **J.-F. Audy, S. D'Amours, N. Lehoux, and M. Rönnqvist.,** Generic Mechanisms for Coordinating Operations and Sharing Financial Benefits in Collaborative Logistics. In L. Camarinha-Matos, X. Boucher, and H. Afsarmanesh, editors, Collaborative Networks for a Sustainable World, volume 336 of IFIP Advances in Information and Communication Technology, pages 537–544. Springer Boston, 2010. ISBN 978-3-642-15960-2.

[4] **J.-F. Audy, S. D'Amours, and L.-M. Rousseau.,** Cost allocation in the establishment of a collaborative transportation agreement-an application in the furniture industry. Journal of the Operational Research Society, 62(6):960–970, 2010. ISSN 0160-5682, 1476-9360.

[5] **G. Biddle and R. Steinberg.,** Common cost allocation in the firm. In H. P. Young, editor, Cost Allocation: Methods, Principles, Applications, pages 31–54. Elsevier Science, Amsterdam, Netherlands, 1985. ISBN 0444878637.

[6] **O. N. Bondareva.,** Some applications of linear programming methods to the theory of cooperative games. Problemy kibernetiki, 10:119–139, 1963.

[7] **F. Cruijssen, O. Bräysy, W. Dullaert, H. Fleuren, and M. Salomon.,** Joint route planning under varying market conditions. International Journal of Physical Distribution & Logistics Management, 37(4):287–304, 2007. ISSN 0960-0035.

[8] **F. Cruijssen, W. Dullaert, and H. Fleuren.,** Horizontal Cooperation in Transport and Logistics: A Literature Review. Transportation Journal (American Society of Transportation & Logistics Inc), 46(3):22–39, 2007. ISSN 0041-1612.

[9] **N. Danloup, H. Allaoui, and G. Goncalves.,** Literature review on or tools andmethods for collaboration in supply chain. In Industrial Engineering and Systems Management (IESM), Proceedings of 2013 International Conference on, pages 1–7. IEEE, 2013.

[10] **Ö. Ergun, G. Kuyzu, and M. Savelsbergh.,** Reducing truckload transportation costs through collaboration. Transportation Science, 41(2):206–221, 2007. ISSN 0041-1655.

[11] **Ö. Ergun, G. Kuyzu, and M. Savelsbergh.,** Shipper collaboration. Computers& Operations Research, 34(6):1551–1560, 2007. ISSN 0305-0548.

[12] **F. Erhun and P. Keskinocak.,** Collaborative supply chain management. In K. G. Kempf, P. Keskinocak, and R. Uzsoy, editors, Planning Production and Inventories in the Extended Enterprise, volume 151 of International Series in Operations Research & Management Science, pages 233–268. Springer US, 2011. ISBN 978-1-4419-6484-7.

[13] **M. Frisk, M. Göthe-Lundgren, K. Jörnsten, and M. Rönnqvist.,** Cost allocation in collaborative forest transportation. European Journal of Operational Research, 205(2):448–458, 2010. ISSN 0377-2217.

[14] **B. Groothedde.,** Collaborative logistics and transportation networks–a modeling approach to hub network design, 2005.

[15] **M. Guajardo and M. Rönnqvist.,** Operations research models for coalition structure in collaborative logistics. European Journal of Operational Research, 240(1):147–159, 2015. ISSN 0377-2217.

[16] **S. Hernández, S. Peeta, and G. Kalafatas.,** A less-than-truckload carrier collaboration planning problem under dynamic capacities. Transportation Research Part E: Logistics and Transportation Review, 47(6):933–946, 2011. ISSN 1366-5545.

[17] **B. Hezarkhani, M. Slikker, and T. Van Woensel.,** On characterization of the core of lane covering games via dual solutions. Operations Research Letters, 42(8): 505–508, 2014. ISSN 01676377.

[18] **X. Hu, R. Caldentey, and G. Vulcano**., Revenue sharing in airline alliances. Management Science, 59(5):1177–1195, 2013. ISSN 0025-1909.

[19] **N. Immorlica, M. Mahdian, and V. Mirrokni.,** Cycle cover with short cycles. InV. Diekert and B. Durand, editors, STACS 2005, volume 3404 of Lecture Notes in Computer Science, pages 641–653. Springer Berlin Heidelberg, 2005.

[20] **A. Kuo and E. Miller-Hooks.,** Developing responsive rail services through collaboration.Transportation Research Part B: Methodological, 46(3):424–439, 2012. ISSN 0191-2615.

[21] **G. Kuyzu.,** Lane covering with partner bounds in collaborative truckload transportation procurement. Manuscript submitted for publication, 2015.

[22] **R. H. Kwon, C.-G. Lee, and Z. Ma.,** An integrated combinatorial auction mechanism for truckload transportation procurement. University of Toronto, http://www.mie. utoronto. ca/labs/ilr/MultiRound. pdf, Working Paper, 2005.

[23] **S. Lozano, P. Moreno, B. Adenso-Díaz, and E. Algaba.,** Cooperative game theory approach to allocating benefits of horizontal cooperation. European Journal of Operational Research, 229(2):444–452, 2013. ISSN 0377-2217.

[24] **E. W. Moore Jr, J. M. Warmke, and L. R. Gorban.,** The indispensable role of management science in centralizing freight operations at Reynolds metals company. Interfaces, 21(1):107–129, 1991. ISSN 0092-2102.

[25] **N. Öner and G. Kuyzu.** Stable coalition selection in collaborative truckload transportation procurement. Working paper, TOBB University of Economics and Technology, 2015.

[26] **O. Ö. Özener and Ö. Ergun.,** Allocating Costs in a Collaborative Transportation Procurement Network. Transportation Science, 42(2):146–165, 2008. ISSN 0041-1655.

[27] **O. Ö. Özener, Ö. Ergun, and M. Savelsbergh.,** Lane-exchange mechanisms for truckload carrier collaboration. Transportation Science, 45(1):1–17, 2011. ISSN0041-1655.

[28] **M. Pal and E. Tardos.,** Group strategyproof mechanisms via primal-dual algorithms. In In Proceedings of the 44th Annual Symposium on Foundations of Computer Science (FOCS, pages 584–593, 2003.

[29] **D. Schmeidler.,** The nucleolus of a characteristic function game. SIAM Journal on applied mathematics, 17(6):1163–1170, 1969.

[30] **L. S. Shapley.,** A value for n-person games. In H. W. Kuhn and A. W. Tucker, editors, Contributions to the Theory of Games II, volume 28 of Annals of Mathematical Studies, pages 307–317. Princeton University Press, 1953.

[31] **L. S. Shapley.,** On balanced sets and cores. Naval research logistics quarterly, 14(4):453–460, 1967.

[32] **C. Vanovermeire and K. Sorensen.,** Integration of the cost allocation in the optimization of collaborative bundling. Transportation Research Part E: Logistics and Transportation Review, 72:125–143, 2014. ISSN 1366-5545.

[33] **C. Vanovermeire and K. Sorensen.,** Measuring and rewarding flexibility in collaborative distribution, including two-partner coalitions. European Journal of OperationalResearch, 239(1):157–165, 2014. ISSN 0377-2217.

CURRICULUM VITAE

Given Name-Surname : Soheyl Zehtabiyan

Nationality : Iranian

Date and Place of Birth : 1988- Maragheh, Iran

E-Mail : Zehtabiyan@gmail.com

EDUCATION BACKGROUND:

- **Bachelor of Science** : 2011, University of Tabriz , Faculty of Mechanical Engineering, Department of Industrial Engineering
- **Master of Science** :2016, TOBB University of Economics and Technology, Department of Industrial Engineering , M.S. in Industrial Engineering

FOREIGN LANGUAGES: English, Azerbaijani, Turkish, Farsi, Arabic

DERIVED PUBLICATIONS AND CONFERENCES FROM THIS THESIS:

- Soheyl Zehtabiyan, Gültekin Kuyzu, 2015., Shipper collaboration with Multiple Coalitions, submitted to the Networks journal.
- Soheyl Zehtabiyan, Gültekin Kuyzu, 2015., Collaborative truckload transportation procurement with multiple coalitions, Vehicle routing and logistics optimization conference (VeRoLog), University of Vienna-Austria, 2015
- Soheyl Zehtabiyan, Gültekin Kuyzu, 2015., Çok Koalisyonlu Tam Kamyon Yükü Gönderici İşbirliği, National Conference for Operations Research and Industrial Engineering (YAEM), Middle East Technical University, Turkey, 2015