RELAY NETWORK DESIGN IN LOGISTICS AND TELECOMMUNICATIONS: MODELS AND SOLUTION APPROACHES

A Dissertation

by

PANITAN KEWCHAROENWONG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2010

Major Subject: Industrial Engineering

RELAY NETWORK DESIGN IN LOGISTICS AND TELECOMMUNICATIONS: MODELS AND SOLUTION APPROACHES

A Dissertation

by

PANITAN KEWCHAROENWONG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Halit Üster
Committee Members,	Brett Peters
	Gary Gaukler
	Tahir Cagin
Head of Department,	Brett Peters

May 2010

Major Subject: Industrial Engineering

ABSTRACT

Relay Network Design in Logistics and Telecommunications:
Models and Solution Approaches. (May 2010)
Panitan Kewcharoenwong,
B.Eng., Sirindhorn International Institute of Technology, Patumthani;
M.A., Chulalongkorn University, Bangkok
Chair of Advisory Committee: Dr. Halit Üster

Strategic network design has significant impacts on the operational performance of transportation and telecommunications industries. The corresponding networks are typically characterized by a multicommodity flow structure where a commodity is defined by a unique origin-destination pair and an associated amount of flow. In turn, multicommodity network design and hub location models are commonly employed when designing strategic networks in transportation and telecommunications applications.

In this dissertation, these two modeling approaches are integrated and generalized to address important requirements in network design for truckload transportation and long-distance telecommunications networks. To this end, we first introduce a cost-effective relay network design model and then extend this base model to address the specific characteristics of these applications. The base model determines relay point (RP) locations where the commodities are relayed from their origins to destinations. In doing this, we explicitly consider distance constraints for the RP-RP and nonRP-RP linkages.

In truckload transportation, a relay network (RP-network) can be utilized to decrease drivers' driving distances and keep them within their domiciles. This can

potentially help alleviate the high driver turnover problem. In this case, the percentage circuitry, load-imbalance, and link-imbalance constraints are incorporated into the base model to control related performance metrics that are affected by the distance constraints. When compared to the networks from other modeling approaches, the RP-network is more effective in controlling drivers' tour lengths and capable of controlling the empty mileage to low levels without adding a large amount of additional travel distance. In telecommunications, an RP-network can be beneficial in long-distance data transfers where the signals' fidelity must be improved/regenerated at RPs along their travel paths. For this setting, we extend the base model to include fixed link setup costs and capacities. From our computational results, our models provide better network configuration that is cost effective and facilitates a better service quality (shorter delays and better connectivity).

Concerning methodology, we develop efficient exact solution algorithms based on Benders decomposition, Lagrangean decomposition, and Lagrangean relaxation. The performance of the typical solution frameworks are enhanced via numerous accelerating techniques to allow the solution of large-sized instances in reduced solution times. The accelerating techniques and solution approaches are transferable to other network design problem settings with similar characteristics.

To *my family*

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Halit Üster, for his invaluable advice and guidance throughout my study at Texas A&M University. Under his supervision, I have learned so many things and greatly appreciate all of his help and encouragement, both professionally and personally. I would also like to thank my advisory committee members, Dr. Brett Peters, Dr. Gary Gaukler, and Dr. Tahir Cagin for providing insightful suggestions and comments throughout the development of this dissertation.

I would like to thank my dear colleagues and officemates, Hui Lin, Joaquin Torres, Homarjun Agrahari, Gopalakrishnan Easwaran, Burcu Keskin, Ezgi Eren, Liqing Zhang, Su Zhao, Xinghua Wang, and Abhilasha Katariya. You have made the lab an enjoyable and a wonderful place to work.

I would like to thank all the officers and volunteers of Texas A&M INFORMS student chapter; it has been fun working with everyone. Moreover, I would like to thank Dr. Üster for being the chapter's advisor and for his helpful advice, and Dr. Peters for supporting our activities. I also would like to extend my appreciation to the departmental staff, Judy Meeks, Claudia Samford, Katerines Edwards, Dennis Allen, Mark Hopcus, and Michele Bork for their help in the chapter's activities.

I am thankful to many of my friends who have made my time in College Station a precious experience of my life. Specifically, I would like to thank Surapong Sirikulvadhana, Yuttapong Jiraraksopakun, Nikornpon Prapaitrakul, Witichai Sachchamarga, Thanathorn Vajirakachorn, and Sakunchai Khumsubdee for their great help and guidance through difficulties.

Finally, I would like to thank my sisters Prangchira and Pajaree, my brother Preetanat, and my parents Prasit and Chamchand, for their endless love and unconditional support. I am extremely proud of my sisters and brother; they have been,

and will continue to be, my inspiration and driving forces for me to go forward . I am wholeheartedly thankful and deeply indebted to my parents, who have sacrificed their whole life in raising me.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

TABLE Page

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The service industry today accounts for almost 55 percent of the total economic activity in the U.S. (U.S. Census Bureau, 2006). Excluding the retail and wholesale areas, the U.S. Census Bureau has categorized the service industry into nine sectors. Among them, two important sectors are 1) information and 2) transportation and warehousing. These two sectors constituted a total of 984.2 billion dollars or 7.4 percent of the total GDP in 2006 (Bureau of Economic Analysis, 2006). In order to emphasize the significance of these sectors, Table 1 summarizes the results from the Service Annual Survey conducted by the U.S. Census Bureau from 2004 to 2006.

Table 1: Revenues and expenses of service industry (U.S. Census Bureau, 2006)

	Revenue/Expense (billions $)			% Change		
Service Type	2006	2005	2004	06/05	05/04	04/03
Information	1048/863	1004/813	995/787	4.4/2.8	5.2/3.2	-/-
Telecom	467/390	446/383	429/372	4.7/1.8	3.9/2.9	-/-
Internet Service	98/81	89/73	82/71	10.2/10.6	7.7/1.9	-/-
Broadcasting	95/72	89/67	83/63	6.7/7.4	6.5/6.4	-/-
Truck Transportation	220/201	207/188	186/170	6.3/7.2	11.1/10.7	10.4/-
Long Distance (LD)	122/113	117/107	105/96	4.3/6.3	11.2/10.8	10.7/-
LD-TL	89/83	85/78	76/70	5.1/6.3	11.5/11.5	10.9/-

In Table 1, the first three columns present the total output in terms of total revenue generated and total expense incurred in each type of service industry. The last three columns present the percentages of change between two consecutive years. In the information sector, telecommunications (TC) accounts for almost half of the total output, while internet service and broadcasting account for another twenty

This dissertation follows the style and format of *Operations Research*.

percent.

In the transportation and warehousing sector, only data corresponding to truck transportation are provided. Although truck transportation does not directly contribute to the U.S. economy as much as the information service sector, the trucking industry is very important to the U.S. economy. In fact, it accounted for 70.69% of the total freight shipment value in 2007 (U.S. Census, 2008). In terms of weight, 60.76% in 2007 (U.S. Deparment of Transportation, 2008) and 68.8% in 2008 (American Trucking Association, 2009) of total freight was shipped by truck. The significance of the trucking industry is expected to continue and the tonnage is estimated to reach 70.9% in 2020 (American Trucking Association, 2009). General purpose trucking can be divided into two main categories: local freight and long distance freight (LD) trucking. LD trucking is composed of 1) the full truckload (TL) trucking industry, and 2) the less-than-truckload (LTL) trucking industry. After viewing Table 1, it is clear that the majority of LD trucking consists of the TL trucking industry. The TL industry has a significant overall impact, as it is the major transportation mode between the manufacturing, retail and wholesale trades. Moreover, TL trucking plays an important role in the transportation of full truckloads between consolidation centers in LTL trucking.

In Table 1, both total revenue and total expense are shown to increase annually due to growing demand and expansion of markets. Growth in total revenue reflects the opportunities for capturing more demand and achieving more profit. At the same time, growth in total expenses indicates an increasing burden on the firms. In fact, total expenses are growing at a faster rate than total revenue in many cases. Therefore, many firms must undergo cost saving programs and initiate more careful management.

I.1. Background

I.1.1. Full Truckload Transportation

Examining the difference between total revenue and total expenses of the TL industry in Table 1, one notes that the profit margins in 2004, 2005, and 2006 were 8.56, 8.56, and 7.33 percent, respectively. This decrease calls for an immediate response to prevent the net margin from further declining. Since the growth of total expenses between 2005-2006 surpasses the growth of total revenue, a plausible response is the reduction of unnecessary expenses. A very high driver turnover rate is one cause of excessive spending in the TL industry.

The turnover problem is costly and influential in the overall performance of TL providers. In addition to driver replacement expenses, other potential impacts include driver shortage, usage of inexperienced drivers, accidents, late deliveries, and customer dissatisfaction. In terms of expenses, driver replacement cost alone is estimated to be around $3000 per driver (Truckload Carriers Association, 2004). More accurate estimation of the turnover cost – which includes the consideration of indirect factors such as 1) entry and exit administration (e.g., training), 2) fixed asset costs due to idle equipment, 3) profit lost due to idle equipment, and 4) insurance and maintenance – is estimated to range from $2000 to $21000 with an average of $8234 (Rodriguez et al., 2000). The overall turnover annual cost is estimated to be around $2.8 billion (for 340000 drivers) in Rodriguez et al. (2000) and $3 billion in Keller and Ozment (1999). Clearly, the total cost of driver turnover constitutes a considerable portion of total industry expenses; reducing driver turnover could yield significant savings. Min and Emam (2003) report different strategies (including monetary approaches such as increased pay, bonus programs, and longevity rewards) of various trucking firms designed to alleviate driver turnover problems. However, none of them has been

verified as an effective approach in the long run.

In TL trucking, a truckload is shipped directly from its origin to its destination by a single driver using a point-to-point (PtP) dispatching approach. After delivering a load to its destination, finding another load for the back haul is generally difficult and an empty direct trip back to the home-base is normally unacceptable. In order to avoid an empty back haul and minimize the empty travel distance, a truck driver is normally assigned to multiple consecutive shipments where the distances between drop-off and pick-up locations are short. However, due to the difficulty of finding a load with its destination near the driver's home-base, the PtP approach usually causes a long tour that keeps the truck driver on the road for an extended period of time and leads to less driver home time. The amount of home time is very important in retaining and recruiting drivers (Min and Lambert, 2002), since 70% of truck drivers quit their jobs because of long tour length (Taylor et al., 1999). Coupled with the poor quality of life on the road, long tour length is the major cause of high truck driver turnover that has occurred over the past several decades.

The TL turnover rate is more than three times the U.S. employment turnover rate of 25.3% in 2007 (Bureau of Labor Statistics, 2007). A driver turnover rate of 85-110% is reported in Mele (1989a,b), 110-120% in Richardson (1994), and remains above 100% (above 80% for smaller TL providers), as shown in Table 2. Although the turnover rate is currently dropping, Schneider Logistics Inc. (2009) reports that the drop has followed the current U.S. economic downturn and holds that it is temporary. In fact, the problem could become even more severe in the next 3-5 years as drivers born in the baby-boomer period (1946-1964) begin to retire (Schneider Logistics Inc., 2009).

The turnover rate is significantly lower in LTL trucking. LTL turnover rate is 10-14.5% in Mele (1989a,b), 14% in 2006-Q3 (American Trucking Association,

Table 2: TL driver turnover rate (%) (Transportation Topics, 2007, 2008)

	2006				2007				2008
Size (annual revenue)	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1
Large TL ($\geq$ 30 million)	116	110	121	121	127	116	113	112	103
Small TL ($<$ 30 million)	111	100	114	112	102	90	87	82	80

2006; American Trucking Association, 2007), and 10% in 2006-Q4 (American Trucking Association, 2007). In general, LTL load size is smaller than a truck's capacity, whereby multiple loads can be consolidated at break bulk terminals (hubs) and transported together for part of their trips to achieve economy-of-scale. To do so, LTL providers utilize two types of truck drivers to operate on the hub-and-spoke networks. In these hub-and-spoke networks, local drivers pick up/deliver loads between origins/destinations and hubs, and lane drivers transfer the loads between any two hubs. This systematic network operation provides truck drivers with more normalized driving schedules, allows them to go home on a regular basis, and eventually leads to a lower driver turnover rate. As a result, the low LTL driver turnover rate validates the use of hub networks to shorten tour length, a practice that could help alleviate the high TL driver turnover. Keller (2002) also strongly suggests truck providers use relay stations and different driver teams to improve drivers' home time and help retain truck drivers.

When applied to the TL industry, a hub does not work as a consolidation center but rather as a switching point that allows truck drivers to relay truckloads before returning to their base stations. Due to the different role of hubs in the TL industry, we represent a hub by the term "Relay Point" and represent a network obtained from locating relay points by the term "Relay Network" throughout this dissertation. In the TL context, the term relay points can be used interchangeably with the terms

transhipment terminals, drop yards, swap points, and hubs. In order to control driving distances, local and lane drivers are permitted to travel no longer than "local and lane tour lengths" from their home base relay points, respectively.

In addition to facilitating regular get-home rates for drivers and helping to alleviate the high turnover problem, operations on the relay network can present other benefits as discussed in Taylor et al. (2001). Among them are 1) an improved truck utilization and, consequently, higher driver utilization, which leads to better compensation for drivers (since drivers are primarily paid based on mileage); 2) the generation of efficient trip schedules and planning facilitated by the assignment of drivers and other workers (maintenance, repair, etc.) to home-base relay points; and 3) the reduction in accidents, training costs and insurance rates due to more experienced drivers with job continuity. The use of relay points can also reduce delivery time by allowing a truckload to continue on its route, with a new driver taking over at a relay point while the previous driver rests before returning with another TL back to his/her home-base. Moreover, reducing the need for on-the-route overnight parking spaces is important, reported in Schneider Logistics Inc. (2009), as truck drivers must travel out-of-route to find legal parking spaces (at a high cost). Such a problem can also be alleviated by having relay points provide additional on-the-route truck stops.

I.1.2. Telecommunications

Relay networks also have numerous applications in the telecommunications and other related industries. Unlike truckload transportation, where the relay network is a potential solution to an existing industry problem, telecommunications have physical limitations in which relay networks are required for their operations.

Long-distance telecommunications involve transmitting signals over a large geographical area where signals normally fade with distance. In order to boost the signal

strength, repeaters or relay points are located to regenerate the signal between the origin and the destination. In wireless applications, the signal sent can travel over only a limited distance and relay points are used to enable long distance connections. Service quality is another important issue to consider in the telecommunication industry. By strengthening the signal at the relay points, there is less opportunity for interrupting noise to enter the signal, hence allowing higher quality service. Relay points can be used to provide alternative communication channels, reduce the traffic on a communication network, and improve network performance. In addition to regenerating/amplifying purposes, in a large telecommunication network, relay points can be switches that must be installed to connect wires with different transmission capacities. Moreover, the relay point can also integrate networks with different technologies (e.g., connect wireless network to optical network).

The design and construction of efficient relay networks are critical for telecommunications operators. According to Deloitte Touche Tohmatsu (2009), the telecommunications industry is currently facing infrastructure competition, both to provide high quality service and to extend the reach to customers. With growing demand, many firms have not only extended their coverage, but also have upgraded their existing copper networks to cable/broadband networks. However, due to the current economic downturn and illiquidity, competing firms are very cautious about extensions and upgrades. Thus, the design of the relay network has become even more crucial for achieving the best possible service with restricted investment.

I.1.3. Motivation and Scope of the Dissertation

This dissertation focuses on applying relay networks to promote better performance of the two industries discussed above, the truckload and the telecommunications industries. Upon observing the current state of these two industries, the motivation

for this research can be summarized as follows:

1. Although full truckload trucking is very important to the U.S. economy, the truck drivers, the essential component of this industry, are not satisfied with their jobs. Monetary incentives may alleviate the problem in the short run; however, without a long term solution, the turnover problem will continue to exist and, very likely, will worsen.

2. With cautious consumer spending, very intense competition, and potentially shrinking profit margins, truckload providers are forced to improve their performance and service quality. Prompt pick-ups, on-time deliveries, shortened shipment time, and reduced number of late shipments will be the keys to customer satisfaction. Because the PtP dispatching randomly assigns drivers over the road, truck providers are in need of a systematic approach to better manage their truck drivers and serve their customers.

3. Demand growth and frequent technology changes require telecommunications firms to continuously adapt and extend their physical networks and operations. Under scarce financial resources, every change in the network is consequential and requires cautious considerations in order to obtain optimal returns on investment.

4. Because multiple telecommunications networks compete across the globe, the infrastructure competition and will continue and require firms to expand their coverage and improve existing networks. Delay, noise interruption, and disconnection are factors affecting the competitiveness of each firm. Thus, the design of the telecommunication network must also take into account the customers' perspective, to not only minimize network construction costs but to also focus

on providing quality services (e.g., high signal quality and minimized delays).

Based on these observations, we believe that these two industries are in need of more efficiently designed physical relay networks. For this purpose, this dissertation aims to capture the important requirements and characteristics of each industry, explicitly address them in effective mathematical models, and develop efficient solution algorithms.

In the next section, we provide a detailed description of the base relay network and its variants, customized to match the different requirements of the truckload and telecommunications industries.

I.2. Relay Network Description

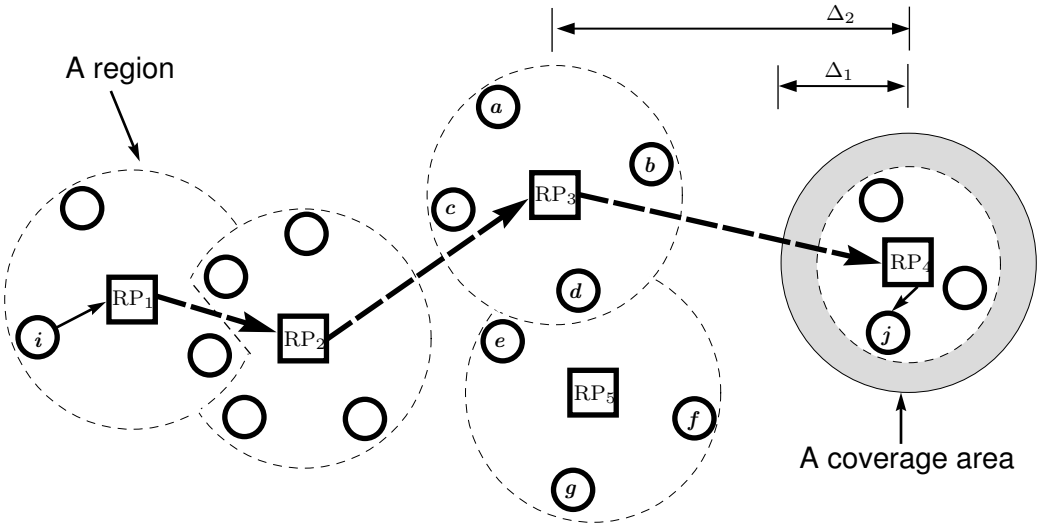
I.2.1. The Base Relay Network Design Problem (RNDP)

We refer to a large geographical service area of a truckload provider or telecommunications operator by a general network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. A set of nodes $\mathcal{N}$ is used to represent the customer and potential relay point locations. In Figure 1, these nodes are represented by circles; some of them have relay points located on them and are represented by squares. Associated with each relay point is the fixed locating cost incurred when a relay point is located, regardless of its utilization level. A set of links $\mathcal{A}$ is used to represent the existence of a connection channel between any two nodes. Associated with each link is the variable link utilization cost that is charged for each unit of demand (truckload or signal) flows on the link.

To represent the demand between node pairs, we let $\mathcal{Q}$ be a set of commodities where each commodity is defined by an origin-destination node pair $[i, j], i, j \in \mathcal{N}$ with a known demand w_{ij} to be transferred from i to j. The located relay points form a relay point-induced network that every commodity must utilize. We make an

assumption that a direct transfer between any two non-relay points without utilizing the relay points network is prohibited. Consequently, there is at least one relay point in the path from a commodity's origin to destination and only the first and the last nodes can be non-relay points (e.g., Figure 1, commodity $[i, j]$ is relayed through RP_1, RP_2, RP_3, and RP_4).

Figure 1: A Schematic View of a Relay Network



There is no special topology requirement of the relay network; however, each non-relay point can only connect to relay points within its Δ_1 distance. Likewise, a connection between any two relay points is allowed only if they are within a Δ_2 distance from each other. A feasible relay network is a connected network of relay and non-relay points formed under the distance requirements that are referred to throughout this dissertation as "distance or tour length constraints".

In order to utilize the relay network, we assume that each non-relay point must be assigned to one unique relay point. This "single assignment" requirement of nodes

forces all of the incoming flow to and the outgoing flow from a non-relay point to pass through the relay point to which it is assigned. Δ_1 can be used to define a coverage area of a relay point, whereas the single assignment requirement is for defining the service region. Specifically, the service region of a relay point is defined by the farthest node(s) that is assigned to the relay point. We also note that a node can be covered by multiple relay points, however, it can be served by only one of them. Thus, the service regions of different relay points are not necessarily the same, in terms of either shape or size. In Figure 1, a solid grey circle represents the coverage area of the associated relay point located at the center. A dashed circle represents a service region of a relay point. Nodes a, b, c, and d are assigned to the relay point RP_3 and nodes e, f, and g are assigned to the relay point RP_5. Node g is covered by two relay points but it is assigned to the relay point RP_3.

In summary, the base "Relay Network Design Problem (RNDP)" considers a given network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, a set of commodities $\mathcal{Q}$, and the restricted distances Δ_1 and Δ_2 to determine:

1. The location of the relay points,
2. The assignment of nodes to relay points, and
3. The actual transfer routes for each commodity

in such a way that the total cost of the relay point location and the cost associated with commodity transfer is minimized.

I.2.2. Relay Network Design in TL Transportation

As mentioned before, the high driver turnover rate in the TL industry is mainly caused by the very long tour lengths that most drivers must endure. In order to address this

tour length issue and potentially alleviate the turnover problem in the TL industry, the use of relay networks can substitute long distance direct shipments with a series of shipments connected at relay points using two types of drivers performing different tasks.

Between non-relay points and relay points, local drivers pick up the shipments from commodities' origins and deliver them to the associated relay points. Then, lane drivers transfer the shipments between relay points over the relay network. Once the shipments arrive at the relay points of the commodities' destinations, another local driver delivers the shipment to the destinations. This systematic framework is similar to operations in the LTL industry. Shorter and more regularized driving routines, along with potentially higher go-home rates, can be achieved, which, in turn, can lead to an improved quality of life for truck drivers. Upon devising the relay network, we would expect a lowered TL turnover rate since TL truck drivers would have similar work descriptions as LTL truck drivers.

To efficiently implement the idea presented above, we extend the base relay network design model to explicitly include three other factors (i.e., load-imbalance, link-imbalance, and link capacity) affecting the operational performance of the relay network.

I.2.2.1. Model 1: RNDP with Load-Imbalance and Percentage Circuitry Constraints

As opposed to the direct shipments in PtP dispatching, the implementation of a relay network trades shortened driving tour lengths with potentially increased empty mileage and additional travel distances.

Empty mileage occurs when the drivers cannot find a load on the first dispatch (forward direction) and on the back haul (backward direction) between the relay

and non-relay points, as well as between any two relay points. In PtP dispatching, only the first dispatch mileage (between a destination and an origin) is empty, since empty back hauls are usually avoided by sacrificing drivers' home times. Due to the difficulty of quantifying the level of empty mileage, there is no evidence verifying that the relay network performs worse than PtP dispatching in terms of empty mileage. However, being able to decrease empty mileage is certainly beneficial to TL providers. In this case, Model 1 considers the control of "load-imbalance", which is defined as the difference between the total incoming and outgoing loads at every service region (Taha and Taylor, 1994; Taylor et al., 1995, 2001; Üster and Maheshwari, 2007). Constructing the service regions (equivalent to assigning nodes to relay points) in such a way that each region has a low level of load imbalance facilitates the opportunity to find loads for back hauls and helps reduce the first dispatch empty miles.

In PtP dispatching, the travel distance is minimized when a shipment is transferred directly from the origin to the destination. On the other hand, shipments in a relay network visit multiple relay points along a possibly more circuitous path. These additional distances are sacrificed in order to better control drivers' distances from their home bases. Since the extra travel distances are related to shipment times, one factor affecting operational performance, Model 1 also considers the control of "percentage-circuitry", or the percentage of additional travel distances, to some acceptable preset upper bounds. Moreover, introducing the percentage circuitry into consideration, can also indirectly help control additional times from connections made at relay points (less connection are made).

I.2.2.2. Model 2: RNDP with Link-Imbalance and Capacity Constraints

The load-imbalance in Model 1 is defined for each relay point and is directly related to the pick-up and delivery operations performed by local drivers. In Model 2, the load-

imbalance is redefined for each pair of relay points and represented by the term "link-imbalance". Specifically, the link-imbalance is the percentage difference between the flow in the forward and backward directions on a link. Contrary to the load-imbalance, the link-imbalance corresponds to the inter-relay point transfer performed by the lane truck drivers. Although the load-imbalance can also be alternately defined (on relay points) to address lane truck drivers' activities (Üster and Maheshwari, 2007), link-imbalance provides a better control of flow balancing since low link-imbalance implies low load-imbalance, but the opposite is not always true. Under the assumption that the drivers always return to their home base relay points after making each delivery, the link-imbalance can directly control empty travel distance. If such an assumption is not made and more complex driving schedules are permitted, then the lane drivers can visit other relay points prior to their return to home base, which allows the TL provider to further reduce the empty travel distance beyond the requirements under the above assumption.

In addition, "link capacity" is another factor that Model 2 explicitly addresses. Capacity is common in most network design problems and, in the TL context, capacity can be in the form of traffic or available workforce. Link capacity is defined as the acceptable upper bound on the total flow in both directions of a link connecting a pair of relay points.

I.2.3. Relay Network Design in Telecommunications

In the telecommunications context, decisions concerning the physical relay network are composed of relay points and links connecting the located relay points. Unlike the previous two models in TL transportation, where the relay points are located on the existing road network, the next two models design the relay network under the assumption that links must be established in advance of permitting the flow between

any two relay points. Moreover, in some cases, the fixed charges must be paid without the physical linkages being built; these cases can also be handled using our models. Specifically, our models have applications in wired (with physical links and fixed charges) and wireless (with only fixed charges) telecommunications.

Although our relay network design problem aims to construct the whole network, with some adjustments in data settings, it can also be used for the purpose of expanding and upgrading existing networks. For example, by fixing the value of decision variables corresponding to the existing relay points and links, the model would return the same relay network with additional relay points and links if their inclusions are beneficial. Moreover, the network links and relay points with high utilization levels are good candidates for upgrades, and those with low utilization levels may possibly be removed.

I.2.3.1. Model 3: RNDP with Fixed Link Construction Cost

In this model, the base RNDP is extended to address the situation when the signal is transmitted through physical linkages established to allow the connections between nodes. We note that the links can be used by the flow in both directions, and their capacity, once established, is abundant (e.g., fiber cables have almost limitless bandwidth (Deloitte Touche Tohmatsu, 2009)). In practice, physical linkages are required for a connection between a non-relay point (an origin or a destination of a commodity) and a relay point, as well as between any two relay points. However, Model 3 only considers the fixed cost associated with the latter case. If links connecting relay and non-relay points must also be established, the associated fixed cost can be embedded into the cost term of the node assignment. Once a node is assigned to a relay point, all the commodities to and from that node must be transmitted through the associated relay point (under the single assignment assumption) and the total

transmission cost (between the non-relay and relay point) now behaves like a fixed charge; thus, the fixed link set-up cost can be included.

I.2.3.2. Model 4: RNDP with Fixed Link Construction Cost and Link Capacity

Model 4 shares many similarities with Model 3 except that the established links now have capacity limitations on the total amount of signal flow through them. This is usually the case in wired and wireless networks where limited bandwidth is shared by multiple commodities. Similar to the previous model, fixed link set-up costs are charged when the links connecting relay points are established. Likewise, link capacity also exists only on the connection between relay points.

In addition, Model 4 can handle the cases where capacity also exists on links connecting relay and non-relay points. Under the single assignment assumption, each non-relay point is connected to a single relay point via a unique link; hence, the link capacity is dedicated only to the commodities that originate from or have the destination at the non-relay point. Consequently, the total capacity requirement of the non-relay point can be predetermined, and the assignment of the non-relay point is restricted only to the links with enough capacity. We note that multiple non-relay points assigned to the same relay point access the relay network through different links that are independent of each other. As a result, they do not share link capacity.

Link capacity can also facilitate traffic management. Bottlenecks occur in high traffic areas, which can consequently lead to signal delay and disconnection. In this case, relay networks with vast connectivity are required in order to ensure high quality service. Such a network can be obtained by using Model 4 with a tight capacity setting. The resulting relay network would contain multiple alternative transmission routes for intensively utilized links; however, they come with additional construction

costs.

I.3. Computational Study

In this dissertation, we present relay network design models customized to match the requirements in the full truckload trucking and telecommunications industries. All models have distinct mathematical formulations that are highly constrained and very large in size, and their unique underlying characteristics make them applicable to different solution approaches. From this observation, we exploit the structure of each model and develop efficient solution algorithms based on Benders decomposition, Lagrangean decomposition, and Lagrangean relaxation frameworks. A variety of computational experiments are conducted to extensively evaluate the performance of our algorithms. Additionally, our computational experiments allow us to examine the influence of problem parameters on the algorithmic performance and characteristics of the resulting relay networks.

All experiments are conducted on Pentium D 3.2GHz workstations with 2GB RAM. Every algorithm is implemented using C++ with STL (Standard Template Library) and Concert Technology (ILOG, Inc.). Whenever the branch-and-cut approach is required, we use CPLEX 9.1 with default settings for cut generation, preprocessing, and upper bound heuristics.

I.3.1. Generation of Test Instances

To serve the objectives discussed above, we generate our test instances in such a way that a wide range of input data and problem parameters is considered. We represent a geographical service area of a TL provider or telecommunications operator using a rectangle with dimensions 150×100 (width × height). We use a set of nodes $\mathcal{N}$

to represent the commodity origins, destinations, and the candidate relay point locations, where $|\mathcal{N}|$ ranges from 20 to 80 nodes. These nodes are uniformly distributed over the 150×100 region. For the cases when the customer locations are concentrated in clusters, we generate the clustered instances by locating 80 nodes over the 150×100 region that is divided into 24 25×25 rectangles. Six of the 24 regions are randomly selected; within each of these regions, 10 nodes are uniformly distributed. The remaining 20 nodes are uniformly distributed over the 150×100 region to provide additional connectivity for these clusters. Note that although we initially assume an interconnected node network, arcs connecting node pairs can be removed if their connections are prohibited; thus, our models do not require the complete node network. Moreover, the connections between node pairs that are too far apart will be addressed by the distance constraints.

In order to generate a set of commodities $\mathcal{Q}$, we first calculate the Euclidean distance between a node to every other node and randomly assign the demand w_{ij} to each node pair using a uniform distribution U[10,20]. The node pairs are then sorted in descending order of their Euclidean distance and divided into three equal sets of long, medium, and short (L-M-S) range demand. We assume that only D percent of the node pairs have demand between them in which the value of D ranges from 20-80 percent. That is, the total number of commodities $|\mathcal{Q}|$ is $|\mathcal{N}|^2\, D/100$ in which $0.6|\mathcal{Q}|$, $0.2|\mathcal{Q}|$, and $0.2|\mathcal{Q}|$ distinct commodities (excluding the node pairs with the same origin and destination locations) are randomly selected from the sets of long, medium, and short range demands. This combination of demands is represented by the 60-20-20 combination. In some experiment settings, we also consider the 20-60-20, 20-20-60, and 40-30-30 combinations. Various data settings considered in our computational studies allow the generation of numerous test instances with many combinations of $|\mathcal{N}|$ and D, as well as the alternative node and demand distributions. We categorized

Table 3: Summary of test instance classes

Node Distribution	$\|\mathcal{N}\|$	Demand Distribution	D			
			20	40	60	80
Uniform	20	60-20-20	Ua1	Ua2	Ua3	Ua4
	25		Ub1	Ub2	Ub3	Ub4
	30		Uc1	Uc2	Uc3	Uc4
	40		Ud1	Ud2	Ud3	Ud4
	60		Ue1	Ue2	Ue3	Ue4
	80		Uf1	Uf2	Uf3	Uf4
	80	20-60-20			Ug3	
	80	20-20-60			Uh3	
	80	40-30-30	Ui1	Ui2	Ui3	
Clusterized	80	60-20-20	Cf1	Cf2	Cf3	
	80	20-60-20			Cg3	
	80	20-20-60			Ch3	
	80	40-30-30	Ci1	Ci2	Ci3	

the generated instances into different problem classes, as summarized in Table 3. Note that all the problem classes with uniform and clustered node distribution have problem class names start with "U" or "C", respectively. For further illustration, the number and the distribution of demands for each instance class are presented in Table 4.

In terms of the cost-based parameters, the fixed cost of locating a relay point is assumed to be 5000 for instance classes with 20, 25 and 30 nodes. Instances with 40, 60, and 80 nodes assume costs of 7500, 10000 and 12500, respectively. Since the number of commodities increases dramatically with increased $|\mathcal{N}|$, we consider increasing the associated fixed cost as $|\mathcal{N}|$ increases to reflect more expensive facilities capable of serving more commodities. On the other hand, the fixed link set-up cost, if it exists, is assumed to be 500 per link. For the variable cost of truckload transportation or signal transmission, we assume a unit transportation/transmission cost. For the

Table 4: The distribution of demands in each problem class

Problem class	$\|\mathcal{N}\|$	D	L-M-S Demand distribution	Number of commodities			$\|\mathcal{Q}\|$
				long	medium	short	
Ua1	20	20	60 - 20 - 20	48	16	16	80
Ua2		40	60 - 20 - 20	96	32	32	160
Ua3		60	60 - 20 - 20	126	48	48	222
Ua4		80	60 - 20 - 20	126	64	64	254
Ub1	25	20	60 - 20 - 20	75	25	25	125
Ub2		40	60 - 20 - 20	150	50	50	250
Ub3		60	60 - 20 - 20	200	75	75	350
Ub4		80	60 - 20 - 20	200	100	100	400
Uc1	30	20	60 - 20 - 20	108	36	36	180
Uc2		40	60 - 20 - 20	216	72	72	360
Uc3		60	60 - 20 - 20	290	108	108	506
Uc4		80	60 - 20 - 20	290	144	144	578
Ud1	40	20	60 - 20 - 20	192	64	64	320
Ud2		40	60 - 20 - 20	384	128	128	640
Ud3		60	60 - 20 - 20	520	192	192	904
Ud4		80	60 - 20 - 20	520	256	256	1032
Ue1	60	20	60 - 20 - 20	432	144	144	720
Ue2		40	60 - 20 - 20	864	288	288	1440
Ue3		60	60 - 20 - 20	1180	432	432	2044
Ue4		80	60 - 20 - 20	1180	576	576	2332
Uf1	80	20	60 - 20 - 20	768	256	256	1280
Uf2		40	60 - 20 - 20	1536	512	512	2560
Uf3		60	60 - 20 - 20	2106	768	768	3642
Uf4		80	60 - 20 - 20	2106	1024	1024	4154
Ug3		60	20 - 60 - 20	768	2106	768	3642
Uh3		60	20 - 20 - 60	768	768	2106	3642
Ui1		20	40 - 30 - 30	512	384	384	1280
Ui2		40	40 - 30 - 30	1024	768	768	2560
Ui3		60	40 - 30 - 30	1536	1152	1152	3840
Cf1	80	20	60 - 20 - 20	768	256	256	1280
Cf2		40	60 - 20 - 20	1536	512	512	2560
Cf3		60	60 - 20 - 20	2106	1024	1024	4154
Cg3		60	20 - 60 - 20	768	2106	768	3642
Ch3		60	20 - 20 - 60	768	768	2106	3642
Ci1		20	40 - 30 - 30	512	384	384	1280
Ci2		40	40 - 30 - 30	1024	768	768	2560
Ci3		60	40 - 30 - 30	1536	1152	1152	3840

distance constraint parameters that occur in all four models, we consider the local and lane tour lengths combinations (Δ_1-Δ_2) of 20-40, 20-50, 30-50, and 30-60. The discussion of the other parameters will be provided in later chapters along with their corresponding models.

I.4. Organization of the Dissertation

The rest of this dissertation is organized as follows. In Chapter II, we introduce the notation that will be used throughout this dissertation. The mathematical formulation of the base RNDP is also presented in this chapter. In Chapter III, we provide a literature review on the hub location problems, the multicommodity network design problems, and applications of relay networks in the full truckload and telecommunications contexts. In Chapters IV and V, we present the mathematical formulations for applications in TL trucking (Models 1 and 2) and telecommunications (Models 3 and 4), respectively. The detailed discussions on the development of the solution algorithm for each model and the extensive computational studies illustrating the algorithmic efficiency are also presented in these chapters. Finally, concluding remarks, the contributions of this research, and future research directions are summarized in Chapter VI.

CHAPTER II

RNDP: THE BASE FORMULATION

In this chapter, we summarize the mathematical notation (decision variables and parameters) for the development of a cost-effective mixed integer programming formulation of the base relay network design model presented in Section I.2.1. To abbreviate the term relay point and relay network, which will be used extensively in this dissertation, "RP" and "RP-network" are used, respectively. Similarly, the term "nonRP" nodes refers to the nodes that do not have relay points located on them.

II.1. Model Parameters

$\mathcal{N}$	Set of nodes, $i, j, k, l \in \mathcal{N}$.
$\mathcal{Q}$	Set of commodities; a commodity is defined by an origin node i and a destination node j with a demand between them, $[i, j] \in \mathcal{Q}$.
w_{ij}	Demand for commodity $[i, j] \in \mathcal{Q}$.
d_{kl}	Distance between node k and node l, $k, l \in \mathcal{N}$.
c_{kl}	Capacity of RP-RP link (k, l), $k, l \in \mathcal{N}$.
T_1	Variable cost between RPs and nonRP nodes per unit demand per unit distance.
T_2	Variable cost between two RPs per unit demand per unit distance.
F_k	Fixed cost of locating an RP at node $k \in \mathcal{N}$.
F_{kl}	Fixed cost of setting up RP-RP link (k, l), $k, l \in \mathcal{N}$.
Δ_1	Permissible distance between a nonRP node and an RP.
Δ_2	Permissible distance between two RP nodes.
Ψ	Permissible level of load-imbalance.
Ω	Permissible level of percentage circuitry.
Θ	Permissible level of link-imbalance.

II.2. Decision Variables

x_{ik} 1 if node i is assigned to an RP at node $k \in \mathcal{N}$, 0 otherwise.

z_{kl} 1 if an RP-RP arc (k,l), $k < l$, $k, l \in \mathcal{N}$ is used in the RP-network, 0 otherwise.

y_{kl}^{ij} Fraction of demand for a commodity $[i,j]$ on an RP-RP arc (k,l). $(0 \leq y_{kl}^{ij} \leq 1)$.

The base model and its four variants can be formulated using these decision variables. Moreover, we note that when x_{ii} is equal to 1, node i is assigned to itself and, therefore, node i is an RP.

II.3. Mathematical Formulation

$$\text{Min} \quad \sum_i \sum_k T_1\, d_{ik} \sum_j (w_{ij}+w_{ji})\, x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} + \sum_k F_k\, x_{kk} \quad (2.1)$$

subject to

$$d_{ik}\, x_{ik} \leq \Delta_1 \qquad \forall\, i, k \in \mathcal{N} \qquad (2.2)$$

$$d_{kl}\, y_{kl}^{ij} \leq \Delta_2 \qquad \forall\, [i,j] \in \mathcal{Q},\ \forall\, k,\ l \in \mathcal{N} \qquad (2.3)$$

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\ \forall\, k \in \mathcal{N} \qquad (2.4)$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \qquad (2.5)$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,\ k \in \mathcal{N} \qquad (2.6)$$

$$y_{kl}^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\ \forall\, k,\ l \in \mathcal{N} \qquad (2.7)$$

$$y_{kl}^{ij} \leq x_{ll} \qquad \forall\, [i,j] \in \mathcal{Q},\ \forall\, k,\ l \in \mathcal{N} \qquad (2.8)$$

$$x_{ik}, \in \{0,1\},\ 0 \leq y_{kl}^{ij} \leq 1 \qquad \forall\, i,\ j,\ k,\ l \in \mathcal{N} \qquad (2.9)$$

The objective function includes all the expenses associated with the implementation of RP-networks, which can be categorized into three main components. The first

component represents the cost of local transportation/transmissions between commodities' origins (and destinations) and RPs. This cost component occurs whenever a nonRP node is assigned to an RP and all the incoming and outgoing flows – from and to the nonRP node – must first travel to the associated RP. For the case that considers the fixed setup cost of links between nonRP nodes and RPs, such cost can be embedded into this local cost component. The second component of the objective function represents the total cost associated with flows transferred between RPs. The combination of the first two components accounts for the annual operation costs from utilizing the RP-network. The last component of the objective function represents the total fixed cost of locating RPs. This cost can be annualized to include the fixed payments during the setup or the construction of RPs, such as the cost of land acquisition, facilities construction, insurance, equipment and tools, and utilities. Fixed costs may be in the form of annual rental fees if firms rent their facilities.

In the constraint set, constraints (2.2) and (2.3) are the distance constraints that restrict the connection between nodes that are farther than Δ_1 and Δ_2 distances, respectively. We note that during the development of our solution approaches, constraints (2.2) and (2.3) will be removed in order to reduce the formulation size. Constraints (2.2) can be removed by setting the value of x_{ik} with the distance d_{ik} greater than Δ_1. Similarly, constraints (2.3) can also be removed by setting the value of y_{kl}^{ij} with the distance d_{kl} greater than Δ_2. However, for the Benders decomposition algorithms, we alternatively remove constraints (2.3) by assigning an arbitrarily large value to d_{kl} if d_{kl} is greater than Δ_2. Constraints (2.4) are the flow conservation constraints defined at each node for each commodity. These constraints define the transfer path – from the commodity's origin, through a set of RPs, to its destination – in which only the origin and destination can be nonRP nodes. If both the origin and destination are assigned to the same RP, then there is only one RP in the transfer

route and thus all $\mathbf{y}$'s are 0. Constraints (2.5) are the single assignment constraints. Each node must be assigned to one unique RP and for nodes that have RPs on their location, they are assigned to themselves. Constraints (2.6), (2.7), and (2.8) provide the structural requirements of the RP-network. Finally, constraints (2.9) state that $\mathbf{x}$ variables are binary while $\mathbf{y}$ variables are real numbers from 0 to 1.

We note that this base formulation is modified from the model presented in Üster and Maheshwari (2007), developed for applications in TL transportation (the differences between these models will be discussed in Chapter IV). We emphasize that this MIP model captures only the general requirement of the base RP-network implementation. In Chapters IV and V, where we concentrate on applications in the TL transportation and telecommunications contexts, we will extend this base model to include the application-specific constraints and modify the base model to match the requirements in each problem accordingly.

Even for the base model without the additional constraints, the formulation's size grows very rapidly with increased $|\mathcal{N}|$ and $|\mathcal{Q}|$. Due to this rapid growth and the corresponding memory requirement, directly solving the base model's formulation with the branch-and-cut approach is very inefficient and limited to small problem instances. Based on this observation, we carefully examine the structure of the base model and four extensions to develop decomposition-based algorithms that allow us to solve significantly smaller problems in an iterative fashion. Specifically, in Chapter IV, we use Benders decomposition and Lagrangean decomposition frameworks to design solution algorithms for applications of RP-networks in TL transportation. Later, in Chapter V, we focus on applications in telecommunications and develop different algorithms based on Benders decomposition and Lagrangean relaxation frameworks.

CHAPTER III

LITERATURE REVIEW

The relay network design problem involves the construction of a relay network and, at the same time, the determination of actual transportation/transmission routes between demands' origins-destinations, in such a way that the total cost is minimized. Considering a modeling approach, our problem is closely related to the "hub location problem" and the "multicommodity network design problem". Therefore, in this chapter, we provide a literature review for both problems, and discuss their relationship to our relay network design problem. Among multiple applications of relay networks, we have chosen to design a relay network for applications in full truckload trucking and telecommunications. Thus, we also review studies related to the use of relay networks in these two areas.

III.1. Hub Location Problem

The hub location problem considers the location of hubs on candidate locations (nodes) and the assignment of non-hub nodes to the located hubs in such a way that the total cost of the hub locations and transportation is minimized. The important assumptions in the hub location problem are:

1. Every commodity must utilize the hub network.
2. The hub-induced subgraph is a complete network.
3. There is a discount on the transportation cost between any two hubs to reflect the economy of scale.

Following the above assumptions, the demand for a commodity from node i must be routed over the hub network before arriving at node j. Due to the complete hub-

induced subgraph and the discount on hub-hub transportation cost, there are at most two hubs on an optimal route from a commodity's origin to its destination. Extensive reviews of the hub location problem can be found in Campbell (1994); O'Kelly and Miller (1994); Campbell et al. (2002); Alumur and Kara (2008).

In the current literature, there are many variants of the hub location problem, each of them posing special characteristics and suitability for different applications. The problem can be capacitated or uncapacitated depending on the available capacity of the hubs. Assuming a very large or unlimited hub capacity, the uncapacitated problem is a special case of the capacitated one. The problem can be either single or multiple allocation depending on whether or not a non-hub node is allowed to access the hub network only through a unique hub. Other variations can be a pre-specified number of hubs to locate with or without the associated fixed charge of locating hubs. A classification of the hub location problem and detailed discussion can be found in O'Kelly and Miller (1994).

In Table 5, we summarize some studies of the hub location problem, categorize them based on the type of capacity and assignment, and note the associated solution methodology developed in each study. Note that the uncapacitated and capacitated variations are represented by "U" and "C", whereas the single and multiple assignment are denoted using "SA" and "MA". Moreover, we refer to the Branch-and-Bound approach as "BB" and Branch-and-Cut approach as "BC".

Among the numerous variations of the hub location problem, the capacitated single assignment version is perhaps the most general model and the most complex to solve. Therefore, in order to illustrate the mathematical models of hub location problems, we provide below the formulation of the capacitated single assignment hub location problem (CSHLP), as presented in Ernst and Krishnamoorthy (1999).

Table 5: Literature for the hub location problem

Paper	Problem	Remark/Methodology
Ernst and Krishnamoorthy (1996)	U-SA	p-hubs; LP-based BB with Simulated annealing heuristics.
Klincewicz (1996)	U-MA	Dual ascent and dual adjustment in BB framework.
Pirkul and Schilling (1998)	U-SA	p-hubs; Lagrangean relaxation with surrogate constraints; Upper bound heuristics.
Abdinnour-Helm and Venkataramanan (1998)	U-SA	Hybrid heuristic between Genetic algorithm and Tabu search
Ernst and Krishnamoorthy (1999)	C-SA	LP-based BB; Simulated annealing and random descent
Ebery et al. (2000)	C-MA	LP-based BB; Shortest path based-heuristics
Mayer and Wagner (2002)	U-MA	Dual ascent BB; Upper bounds from complementary slackness and improved heuristics
Marin (2005)	C-MA	LP-based BB; Re-allocation heuristic.
Wagner (2007)	U-SA	Locate one hub in each cluster; Constraint programming.
De Camargo et al. (2008)	U-MA	Benders decomposition with multiple cuts and ε-opt framework
Yoon and Current (2008)	U-MA	Fixed arc cost; Dual-based heuristic
Rodríguez-Martín and Salazar-González (2008)	C-MA	BC based on Benders and double decomposition; LP-based heuristic with local search algorithms.
Randall (2008)	C-SA	Ant colony heuristics
Silva and B. (2009)	U-SA	Multi-start Tabu search and Two-stage Tabu search.
Contreras et al. (2008)	C-SA	Lagrangean relaxation; Local search heuristics.

Parameters

$\mathcal{N}$ Set of nodes, $i, j, k, l \in \mathcal{N}$.

d_{ij} Distance between node i and j, $i, j \in \mathcal{N}$.

χ Unit transportation cost between node and hub.

δ Unit transportation cost between hub and node.

α Discounted transportation cost between any two hubs.

w_{ij} Flow between node i and j, $i, j \in \mathcal{N}$.

O_i $\sum_j w_{ij}$, $i \in \mathcal{N}$.

D_i $\sum_j w_{ji}$, $i \in \mathcal{N}$.

Γ_k Capacity of hub $k, k \in \mathcal{N}$.

F_k Fixed cost of locating hub at node $k \in \mathcal{N}$.

Decision variables

x_{ik} 1 if node i is allocated to hub at node k, $i, k \in \mathcal{N}$, 0 otherwise.

($x_{kk} = 1$ implies hub at node i)

y^i_{kl} Total flow of commodities from i that is routed between hubs k and l, $i, k, l \in \mathcal{N}$.

Model formulation

$$\text{Min} \quad \sum_i \sum_k d_{ik}\, x_{ik}(\chi\, O_i + \delta\, D_i) + \sum_i \sum_k \sum_l \alpha\, d_{kl}\, y^i_{kl} + \sum_k F_k\, x_{kk} \tag{3.1}$$

subject to

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{3.2}$$

$$x_{ik} \le x_{kk} \qquad \forall\, i, k \in \mathcal{N} \tag{3.3}$$

$$\sum_i O_i\, x_{ik} \le \Gamma_k\, x_{kk} \qquad \forall\, k \in \mathcal{N} \tag{3.4}$$

$$\sum_l y^i_{kl} - \sum_l y^i_{lk} = O_i\, x_{ik} - \sum_j w_{ij}\, x_{jk} \qquad \forall\, i, k \in \mathcal{N} \tag{3.5}$$

$$x_{ik} \in \{0, 1\},\ y^i_{kl} \ge 0 \qquad \forall\, i, k, l \in \mathcal{N} \tag{3.6}$$

In the objective function (3.1), the first two terms correspond to the total transportation cost. The first term represents the transportation between non-hub nodes and hubs, calculated using the non-discounted transportation cost. The second term represents the inter-hub transportation where the unit transportation cost is discounted ($\alpha \leq \chi$ and δ) due to the economy of scale. The last term in the objective function accounts for the fixed cost associated with the hub locations. Constraints (3.2) enforce the single assignment of non-hub nodes to hubs. Constraints (3.3) ensure that nodes can be assigned only to hubs. Constraints (3.4) impose the capacity limitation on hubs. Constraints (3.5) are flow conservation constraints. Finally, constraints (3.6) state that x_{ik} are binary and y_{kl}^{i} are nonnegative real numbers. Ernst and Krishnamoorthy (1999) also provide a tighter alternative formulation that utilizes four indices of decision variables. However, due to its compact formulation size that requires significantly smaller memory, the three indices formulation (3.1)-(3.6) is preferable and is utilized in later studies. In addition, we note that for the uncapacitated problem, constraints (3.4) are removed.

The hub location problem is widely acknowledged among researchers; however, its complete hub-induced subgraph assumption can be impractical for some situations. Upon observing this, Campbell et al. (2005a,b) introduced new models, "hub arc location problems", to address the hub-related problem without the complete subgraph assumption. Specifically, the hub arc location problem considers locating a fixed number of hub arcs on which the unit transportation cost is discounted and both of the arcs' ends imply hubs. Four special cases are discussed, whereby an enumeration based algorithm is developed for each case.

Other interesting variations of the hub location problem are the "hub covering problem" and "p-hub center problem". Both problems are based on the same problem settings as the typical hub location problem such that 1) there exists the discount for

hub-hub transfer from economy of scale and 2) all the origin and destination nodes must be assigned to hubs. Specifically, the hub covering problem considers minimizing the number of hubs required to cover all the nodes in such a way that the distance constraints are satisfied. The distance constraints can be alternatively defined for 1) the paths connecting origin-destination pairs (though the hub network), 2) the links connecting the origins or destinations to the hubs to which they are assigned, or 3) the links connecting a pair of hubs.

On the other hand, the p-hub center problem involves locating p hubs in such a way that the maximum distance between the origins or the destinations to the hubs is minimized. The objective can be altered to include the maximum distance between hub-hub connections or to consider minimizing the maximum distance between origin-destination pairs. The node assignment can be either single or multiple allocation, similar to the typical hub location problem. The formulation of the single and multiple allocation hub covering problem can be found in Wagner (2008) and of the p-hub center problem in Ernst et al. (2009).

III.1.1. Relationship with Our Models

Although our base model is closely related to the uncapacitated single assignment hub location problem (USHLP), they differ significantly in many aspects. Most importantly, the complete hub-induced subgraph assumption, which limits the solution space to containing only the paths with at most two hubs, is removed due to the inclusion of distance constraints. In fact, these distance constraints permit transportation to take place only between two locations that are not too far apart (transportation routes can consist of multiple transfer locations), which can be more practical in the context of freight transportation and telecommunications. In addition to the distance constraints, all of our models impose special requirements that further complicate the

base model, making it further differ from the hub location problem. If the node assignments are given, the transportation route in USHLP is readily known because of the complete subgraph assumption, whereas the subsequent problems of determining the transportation route in our models (including the base model) are still complicated and require further calculations.

We also note that the base model is a generalization of USHLP; both problems are similar if the distance constraints are removed. The flow conservation constraints (3.5) are defined in aggregated form for all the commodities that originate from a node i, $i \in \mathcal{N}$. On the other hand, our flow conservation constraints (2.4) are defined for each commodity. Moreover, constraints (3.7) given below can be included in the base model in order to address the hub capacity as CSHLP.

$$\Big(\sum_{i} (w_{ij} + w_{ji})\, x_{ik} + \sum_{i} \sum_{j} \sum_{k} \sum_{l} w_{ij}\, y_{kl}^{ij} \Big) \leq \Gamma_k\, x_{kk} \quad \forall\, k \in \mathcal{N} \tag{3.7}$$

Constraints (3.7) are the capacity constraints defined on every node where the LHS represents the total flow allowed to transfer through node k, only if an RP is located there. After including constraints (3.7) and relaxing the distance constraints, as discussed above, the base model will serve the same objective as CSHLP. However, we note that testing this variation of the base model is beyond the scope of our study, and we leave it for future study.

Additionally, our four models are related to the hub covering and p-hub center problems due to the distance constraints and the local and lane tour lengths. The local tour length (Δ_1) controls the distances between the origin and destination nodes to relay points, while the lane tour length (Δ_2) controls the distances between any two relay points. Moreover, Model 1 includes the percentage circuitry that controls the distance between the origin and destination node pairs. If the total transporta-

tion/transmission cost term in the objective function of our models is removed, our problems are similar to the hub covering problem, and the resulting problems would try to minimize the number of relay points located. On the contrary, if the number of relay points to locate is given, then our problems are similar to the p-hub center problem as the problem would then minimize the transportation/transmission costs, hence minimizing the total distances (origins-relay points, between relay points, and relay points-destinations).

III.2. Multicommodity Network Design Problem

The multicommodity network design problem (MND) considers two decisions: 1) network construction decisions and 2) commodity routing decisions. In order to construct a network, a set of arcs must be established so that there exists a path from the commodities' origins to destinations. Note that transportation between any two locations is allowed only if the arc connecting them has been established in advance. The objective is to minimize the total cost of the network construction and the routing. Particularly, MND directly addresses the trade off between the additional cost from setting up more arcs and the savings in routing cost resulting from increased numbers of potential origin-destination paths.

There are two main variations of this problem, capacitated and uncapacitated MND. The main difference lies in the existence of the arc capacity limitation of the total flow and the problem is capacitated if such a limitation exists; otherwise, the problem is uncapacitated. An extensive review of the capacitated MND literature can be found in Balakrishnan et al. (1997). Later, Costa (2005) provides a review on both the capacitated and uncapacitated MND; however, the review is limited to only those applied to Benders decomposition. Both problems provide interesting research

areas and have received considerable attention from research communities. Numerous solution approaches have been customized to solve MND problems, either exactly or heuristically. For illustration purposes, Table 6 provides a short summary of studies in the MND area. In Table 6, the entries "U" and "C" in the second column indicate whether the problem is uncapacitated or capacitated, respectively.

Clearly, we can see in Table 6 that current attention is being directed to the capacitated version of the problem. Due to the capacity limitation, capacitated MND poses a complicated and challenging problem structure for the development of solution algorithms. Moreover, it also provides an excellent test bed for the comparison between different methodologies, especially for heuristic algorithms. For later discussion regarding the relationship to our problem, we provide a mathematical formulation of the capacitated MND below. This formulation is presented in Gendron et al. (1998). Gendron et al. (1998) define the commodity set using $\mathcal{K} = \{1, \ldots, k\}$, however, for consistency in notation, we represent the commodity set using $\mathcal{Q}$ where $[i, j] \in \mathcal{Q}$ is the origin-destination node pair with demand w_{ij} between them. The other parameters and decision variables are redefined using the same or similar notation as provided in Chapter II.

Table 6: Literature for the multicommodity network design problem

Paper	Setting	Remark
Magnanti et al. (1986)	U	Benders decomposition with strong/pareto cuts.
Balakrishnan et al. (1989)	U	Dual-ascent with add-drop dual-based heuristics.
Holmberg and Hellstrand (1998)	U	Lagrangean heuristics in BB
Gendron et al. (1998)	C	Compare different Lagrangean relaxations; Resource-decomposition heuristics.
Holmberg and Yuan (2000)	C	Lagrangean heuristics in BB; Develop new cutting criteria.
Crainic et al. (2000)	C	Path-based formulation; Tabu the flow variables in simplex pivot.
Crainic et al. (2001)	C	Compare different Lagrangean relaxations; Subgradient and Bundle-based optimization.
Crainic and Gendreau (2002)	C	Parallel Tabu search sharing data to and from the pool of solutions.
Ghamlouche et al. (2003)	C	Cycle-based neighborhood in Tabu search.
Ghamlouche et al. (2004)	C	Ghamlouche et al. (2003) with Path-relinking.
Crainic et al. (2004)	C	Apply Lagrangean perturbation and long term memory in Slope scaling heuristics.
Alvarez et al. (2005b)	C	Different Scatter searches.
Alvarez et al. (2005a)	C	Alvarez et al. (2005b) with GRASP.
Crainic et al. (2006)	C	Parallel cycle-based Tabu search sharing data across consecutive levels; Level is the number of tabu arcs.
Belotti et al. (2007)	C	Step function for node fixed cost; BB with two valid inequalities.

Parameters

$\mathcal{N}$ Set of nodes, $i, j, k, l \in \mathcal{N}$.

$\mathcal{A}$ Set of arcs, $(k, l) \in \mathcal{A}$.

$\mathcal{Q}$ Set of commodity, $[i, j] \in \mathcal{Q}$.

w_{ij} Demand for commodity $[i, j] \in \mathcal{Q}$.

T_{kl}^{ij} Unit transportation cost for commodity $[i, j]$ on arc (k, l).

F_{kl} Design cost for arc (k, l).

c_{kl} Capacity of arc (k, l).

b_{kl}^{ij} $= \min\{w_{ij}, c_{kl}\}$.

Decision variables

z_{kl} 1 if arc (k, l) is designed, $(k, l) \in \mathcal{A}$, 0 otherwise.

y_{kl}^{ij} Flow of commodity $[i, j]$ on arc (k, l), $[i, j] \in \mathcal{Q}, (k, l) \in \mathcal{A}$.

Model formulation

$$\text{Min} \quad \sum_{[i,j]\in\mathcal{Q}} \sum_{(k,l)\in\mathcal{A}} T_{kl}^{ij}\, y_{kl}^{ij} + \sum_{(k,l)\in\mathcal{A}} F_{kl}\, z_{kl} \tag{3.8}$$

subject to

$$\sum_{l\in\mathcal{N}/\{k\}} y_{kl}^{ij} - \sum_{l\in\mathcal{N}/\{k\}} y_{lk}^{ij} = \begin{cases} w_{ij}, & k = i \\ -w_{ij}, & k = j \\ 0, & \text{o.w.} \end{cases} \qquad \forall [i, j] \in \mathcal{Q} \tag{3.9}$$

$$\sum_{[i,j]\in\mathcal{A}} y_{kl}^{ij} \le u_{kl}\, z_{kl} \qquad \forall (k, l) \in \mathcal{A} \tag{3.10}$$

$$y_{kl}^{ij} \le b_{kl}^{ij}\, z_{kl} \qquad \forall (k, l) \in \mathcal{A}, [i, j] \in \mathcal{Q} \tag{3.11}$$

$$y_{kl}^{ij} \ge 0 \qquad \forall (k, l) \in \mathcal{A}, [i, j] \in \mathcal{Q} \tag{3.12}$$

$$z_{kl} \in \{0, 1\} \qquad \forall (k, l) \in \mathcal{A} \tag{3.13}$$

In the objective function (3.8), the first term represents the total routing cost of

every commodity and the second term represents the network construction (arc selection) cost. Constraints (3.9) are the flow conservation constraints. Constraints (3.10) ensure that the total flow on any arc does not exceed the arc capacity. Constraints (3.11), although redundant, are included in many studies (Gendron et al., 1998; Holmberg and Yuan, 2000; Crainic et al., 2000) to improve the lower bounds' quality. Finally, constraints (3.12) state that y_{kl}^{ij} are positive real numbers and constraints (3.13) impose the binary requirement of z_{kl}.

For an uncapacitated problem, constraints (3.10) and (3.11) can be replaced by constraints (3.14), however w_{ij} is now 1 for every commodity and T_{kl}^{ij} are the cost of transporting the whole demand of commodity $[i, j]$ on arc (k, l). This uncapacitated formulation is presented in Holmberg and Hellstrand (1998).

$$y_{kl}^{ij} \leq z_{kl} \quad \forall (k,l) \in \mathcal{A}, [i,j] \in \mathcal{Q} \tag{3.14}$$

The arc-based formulation has been widely used; however, another stream of research focuses on path-based alternative formulation, especially for the case when a commodity must be routed only on a single path. Both formulations are equally good for the capacitated problem in terms of LP lower bound strength; however, for the uncapacitated case, the arc-based formulation has tighter LP bounds (Rardin and Choe, 1979), as cited in Gendron et al. (1998).

III.2.1. Relationship with Our models

Comparing our problem to the MND problems, we found that Model 3 and Model 4 (presented in Chapter V) have close relationships with the uncapacitated and capacitated MND, respectively. The differences between Model 3 and the uncapacitated MND (and also between Model 4 and capacitated MND) are the existence of distance constraints, the location of RPs, and the single assignment of nodes. For given RP

locations and node assignments, our problems reduce to MND with distance constraints. More specifically, if the distance constraints are relaxed and the fixed cost of locating RPs is set to zero, then all nodes imply RPs (as it is free to locate RPs) and every node is assigned to itself; thus, Models 3 and 4 are the same as the uncapacitated and capacitated MND. Following this observation, we can conclude that Models 3 and 4 are generalizations of MND problems where the single assignment and the distance constraints (and also the location of RPs) provide special topological requirements that further complicate the problem. We note that, with only minor adjustments, our solution algorithms can be directly applied to solve both types of MND.

It is also interesting that, although Models 1 and 2 do not consider arc set-up, their application-specific constraints (namely, load-imbalance and link-imbalance constraints) are similar to some studies in the area of capacitated MND. Pedersen et al. (2009) introduce "asset-balance constraints" that are incorporated into the capacitated MND. These constraints require the balanced location of arcs in such a way that the number of arcs (without considering flow on them) entering (in-degree) and leaving (out-degree) a node must be equal. Likewise, the load-imbalance constraints in Model 3 require a balanced flow of loads entering and leaving a relay point, while the link-imbalance constraints in Model 4 consider a balanced flow between a pair of RPs in forward and backward directions. We also note that the link-imbalance constraints in Model 4 have not been considered before.

III.3. Truckload Applications (TL)

The use of relay points (called hubs in early studies) to shorten tour lengths and help alleviate the high driver turnover problem in the full truckload trucking industry has

been extensively examined in various simulation studies.

Taha and Taylor (1994) develop a rule-based simulation to determine the number, location, and service area of the hubs, and to suggest when to perform a direct shipment. The results show that, through the use of hubs, the TL provider can trade off the reduction in tour length with extra travel distances (circuitry) and the first dispatch empty miles. Also with simulation tools, Taylor et al. (1995) examine network scenarios with different hub locating methodologies, number of hubs, and a driver's permissible number of hubs from home base. The operations of truck drivers follows those of the less-than-truckload trucking. Specifically, local drivers pick up and deliver truckloads between hubs and non-hub locations, lane drivers transfer truckloads between any two hubs, and non-network drivers perform direct shipments. The results suggest the construction of a service area with a low "load-imbalance" level – the difference between the total incoming and outgoing load – to achieve good network performance.

Taylor et al. (1999) simulate a variety of dispatching methods that utilize a zone model, a key hub, a key lane, and point-to point (PtP) dispatching. The results show that the zone model performs best in terms of empty miles, miles per driver per day, and percentage of late loads, but it may cause a high level of circuitry. Taylor and Meinert (2000) conduct extensive simulation experiments for a special case including two rectangular adjacent zones. The results show that the number of hubs and the radius of the zone (tour length) significantly impact the performance of the model and it is observed that the use of hubs compares favorably to PtP dispatching in reducing tour lengths and flow times (through relaying opportunities at hubs). Taylor et al. (2001) introduce a variety of zone models with different number of hubs, zones, interior points, and different levels of permissible circuitry and load-imbalance. An assumption is made that a TL can be relayed only once on its trip. The results show

that when zones are created in such a way that the load imbalance level is low, the minimum tour length can be achieved with only a small increase in the total flow time.

All of the simulation studies discussed above provide supportive evidence for the use of hub-networks in reducing truck drivers' tour lengths. Clearly, the tour length allowances and the load-imbalance level should be carefully controlled in order to obtain a good network. The only drawback is the extra travel distance (circuitry) that must be taken into consideration, from an operational cost effectiveness perspective.

These simulation studies differ from our models in a number of ways. While all these studies compare the performance of pre-determined network configurations (the numbers and locations of hubs), our models seek the best configuration endogenously (in Chapter IV, we conduct an experiment to compare the performance of networks obtained using their hub location strategies with that from our Model 1). The hub location strategies in these simulations are either based on the volume of freight or the level of load imbalance; our models locate RPs on the candidate locations only if the tradeoffs between the fixed RP locating costs and the transportation cost savings are profitable. Most importantly, the candidate networks in the above simulation studies are evaluated using performance indices such as driver tour lengths, empty mileage, extra travel distance, percentage of late loads, and flow time. Our models explicitly control driver tour lengths and extra travel distance to satisfactory levels and consider minimizing the total cost of locating RPs and transportation. Moreover, since the total travel distance and flow time are directly proportional to the transportation cost, they are implicitly minimized in our models.

Besides the simulation studies discussed earlier, there are other studies that utilize relay points in reducing tour length. Under the same assumptions that 1) there is no fixed cost associated with locating RPs and 2) RPs can be located anywhere

on the network, Hunt (1998) and Ali et al. (2002) develop different algorithms to locate RPs on the U.S. highway system. The algorithm developed by Hunt (1998) constructs the RP-network in three steps. First, the algorithm solves the shortest path problem for each commodity. Then, in the second step, RPs are located along the shortest path by the "spring algorithm" or, alternatively, by a greedy algorithm. Finally, the commodities are re-routed on the constructed RP-network to obtain their actual transportation paths. The results show that both the tour length and the flow time can be greatly reduced; however, some commodities may have a high level of circuitry, in which case, a direct shipment is suggested.

On the other hand, Ali et al. (2002) develop three iterative approaches to locate a minimum number of RPs on a network, while satisfying a distance constraint. The first approach iteratively locates RPs along the shortest path between the origin and the destination of each commodity, and loads are restricted to travel on this path. The second approach allows the load to exit from the shortest path route at some intersection to utilize the previously located RPs. The load must then return to the same intersection before continuing to travel on the shortest path. The third approach permits the transportation of load on any path that has an additional distance within some permissible value. The results show that a minimum number of RPs is required if the commodities are routed in ascending order of their shortest path lengths in the first approach, which is the same number in the second approach. For the third approach, the number of RPs depends on the permissible additional distance. We note that neither of the two provides a mathematical formulation to benchmark their algorithms.

Recently, Üster and Maheshwari (2007) have derived a mathematical model for the construction of RP-networks that consider the tour length, load-imbalance, and percentage circuitry constraints. The objective function is to minimize the total cost

associated with locating RPs and transportation. Note that Model 1 is based on this model and we provide a comparison between these two models in Chapter IV. They analyze the impact of the parameters and provide an efficient tabu search heuristic utilizing exchange, add, and drop neighborhoods. The results show the inter-relationship between the parameters that lowered load-imbalance level can be obtained by increasing the local and lane tour length, while improved percentage circuitry level can be achieved by increasing the lane tour length and decreasing the local tour length.

III.4. Telecommunications Applications

Relay networks have also been applied to applications in telecommunications and related industries (e.g., internet and broadcasting). The most notable study related to our problem is by Cabral et al. (2007). In a telecommunications tree network (single origin-multiple destinations), repeaters (RPs) must be located to amplify signal quality with respect to a distance constraint. The construction of the RP-network includes the location of RPs and the set up of transmission links, with the objective being to minimize the total RP-network construction cost. Four heuristic algorithms are developed and benchmarked with a lower bound obtained from solving a path-based formulation with column generation. Due to several similarities between this model and our Model 3, we make the comparison between the two models in Chapter V.

Based on their previous work, Cabral et al. (2008) have developed a two-step method for the design of a wide area broadband internet network in Alberta. The first step employs the algorithms in Cabral et al. (2007) to determine the network structure that consists of shelter (RP) locations and links. The second step uses tabu search heuristics to determine the type of optical fiber to install on each link and the location of repeaters and switches (for connecting links with different tech-

nologies/types) on the located shelters. The objective is to minimize the total cost of technology installation and location of repeaters and switches in such away that the total delay (induced from repeaters and switches) for each commodity does not exceed a permissible level.

In the context of an internet broadband wireless network, So and Liang (2006) construct the RP-network on a complete graph to connect a base station to end users who are scattered in a large service region. The objective is to minimize the number of RPs and the penalty of unmet demand in such a way that the total flows between the end users and RPs do not violate the RP capacity. Benders decomposition algorithm is used to solve this model, and the results show that fewer RPs are required with increased RP capacity. However, a minimum number of RPs cannot be further reduced. This problem is significantly different from our models in many ways. While their model only concerns the minimum number of RPs, our models consider the trade-off between the total cost of RP locations and the transmission cost savings that arise from locating more RPs. Moreover, our models do not consider the complete graph assumption due to the distance constraints and our models involve multicommodity flow instead of one-to-many demand.

Kashyap et al. (2006) consider minimizing the congestion of an existing wireless backbone network by creating alternative bypass channels formed by a series of RPs. In this model, each node has a limitation on the number of channels it can connect, and if the bypass distance is beyond a single transmission range, then a sequence of RPs is required. To do this, the additional edges (formed by using RPs) are located on an existing network using three rule-based greedy algorithms. Additionally, to improve the quality of solutions, a "rollout" algorithm is applied to modify the solutions based on their future expectations. Finally, maximum congestion is obtained by solving the network flow problem on the new networks. The results show that maximum

congestion can be reduced when locating the first RP and the reduction gradually decreases as more RPs are located. The results also demonstrate the capability of the rollout algorithm to improve the solutions from all greedy algorithms.

III.5. Positioning in the Current Literature

Having reviewed the literature in the areas discussed above, we observe that this dissertation research can be positioned in many different research areas:

1. **Hub Location Problem Literature**: Models 1-4 are closely related to the uncapacitated single allocation hub location problem (USHLP) without the complete hub-induced subgraph assumption. Our problems also integrate the key characteristics of the hub center problem into the USHLP by having the distance constraints control the maximum distances between non-relay and relay points, and between relay point pairs, within some permissible levels. In addition, Model 1 also contains the percentage circuitry constraints that help control the total distance of the paths between the origin-destination pairs. In summary, our problems are not only related to numerous variations of the hub location problems, but can be considered as integrating these variations with more flexibility in modeling real problems. Additionally, the application specific constraints (i.e., load-imbalance, link-imbalance, and capacity constraints) further complicate our problems.

2. **Multicommodity Network Design Problem Literature**: Compared to the multicommodity network design problem (MND), Model 3 and Model 4 are related to the uncapacitated and capacitated MND, respectively. In both cases, our models are more constrained due to the location of RPs, distance constraints, and single assignment constraints.

Considering two areas discussed above (items 1 and 2), our Model 3 (and Model 4) integrates the key characteristics of both USHLP (location of RPs and single assignment) and MND (arc selection) into one general model that is more flexible in capturing different requirements in some applications.

3. **Solution Methodology**: We develop efficient solution algorithms based on Benders decomposition, Lagrangean relaxation, and Lagrangean decomposition. For Benders decomposition, we successfully enhance their performance through the use of strong Benders cuts, cut disaggregation schemes, ε-Optimal, and surrogate constraints. A new method for obtaining the strengthened Benders cuts is also introduced. For Lagrangean decomposition, we duplicate decision variables in aggregated forms, which facilitates the reduction of formulation size and the decomposition of the relaxed problem. For Lagrangean relaxation, we derive surrogate constraints for improving Lagrangean relaxation lower bounds. In all cases, we develop improvement heuristics and apply the heuristic solution to enhance the algorithmic performance.

4. **Applications in Truckload Logistics**: For the TL applications, we introduce a framework that can potentially alleviate the high driver turnover problem. By resembling the operation in less-than-truckload (LTL) trucking, we propose that truck providers relay their shipments, as in the LTL industry, instead of making long direct shipments. By doing this, we expect a reduction in turnover rate since TL truck drivers now perform very similar tasks to LTL drivers (LTL trucking has a very low turnover rate). Thus, the RP-network is our proposed potential solution to the existing industry problem. In addition, we also provide mathematical models for the design of a cost effective relay network as opposed to comparing specific scenarios using simulations. The mathematical models, in

turn, allow us to incorporate important operational efficiency constraints, such as tour length, circuitry, load-imbalance, and link-imbalance constraints, then control them within permissible levels.

5. **Applications in Telecommunications**: In the telecommunications area, we introduce a new general model that includes RP locations, arc selection, and routing decisions under capacitated and uncapacitated arc settings. The applications are found in both wireless and wired telecommunications. Moreover, our models are applicable to the hybrid wired-wireless telecommunications networks, which have been receiving increasing attention (Sarkar et al., 2009). An example of this hybrid network can be found in Sarkar et al. (2009), where the model considers an optical fiber network, where each end user has a wireless connection to only a single optical unit. Models 3 and 4 can be directly used for the construction of such networks. More specifically, the optical network can be represented by RP-network and RP-RP links, while the single wireless connection to end users can be represented by a single assignment of nodes.

CHAPTER IV

RELAY NETWORK DESIGN FOR TRUCKLOAD TRANSPORTATION

In this chapter, the base relay network design model is extended to match the requirements in truckload (TL) transportation. The construction of a strategic relay network (RP-network) takes into account the operational issues in the TL transportation context, such as empty mileage, percentage circuitry (additional travel distance), and capacity limitation. For the empty mileage, the load-imbalance and the link-imbalance constraints are designed to balance the flow of truckloads and help control the empty travel distance. On the other hand, percentage circuitry and capacity limitation can be expressed in mathematical form and are controlled using the percentage circuitry and capacity constraints, respectively. Although it is possible to incorporate all these constraints into one model, it would be extremely difficult to generate test instances that are feasible with respect to all these constraints at the same time. Therefore, we handle these constraints using two models, each with two types of constraints. However, if all four constraints must be considered at the same time, we can incorporate the other two constraints into the objective function so that the violation of these constraints is penalized in the form of additional cost. After making this modification, the modified model (two constraints in the constraint set and two constraints in the objective function) can be solved using the solution algorithms for the model with the two types of constraints developed in this chapter.

Closely examining the requirement of each type of constraint, we observe that the load-imbalance constraints should be addressed at the same time as the percentage circuitry constraints. To achieve a small level of load-imbalance, many non-relay points may be assigned to a relay point. As a result, many commodities must then travel on circuitous paths, which would lead to increased transportation cost and

time. Therefore, including the percentage circuitry constraints would be beneficial in preventing such incidents. Moreover, we also observe that the capacity and the link-imbalance should be addressed in the same model. While trying to balance the flow in both directions, the link-imbalance constraints could potentially send a large amount of flow on some links, which may lead to network congestion. Thus, the incorporation of capacity constraints would help control the total amount of flow.

Based on the above observations, we consider addressing these constraint sets separately, using two mathematical models. In Section IV.1, Model 1 considers the load-imbalance and the percentage circuitry constraints in addition to the general requirements of the relay network design. In Section IV.2, the link-imbalance and the capacity constraints are incorporated into Model 2.

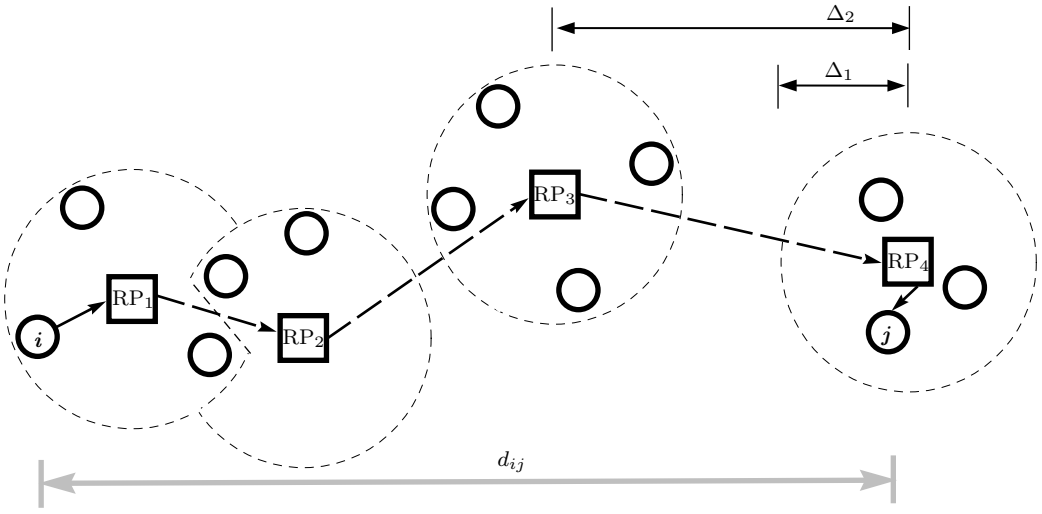
IV.1. Model 1: RNDP with Load-Imbalance and Percentage Circuitry Constraints

The operational characteristics of the RP-network in Model 1 are similar to those in the base model. We consider a large geographical service area of a TL provider represented by an underlying road network in which a set of locations/nodes $\mathcal{N}$ – which can be the commodities' origins or destinations, as well as potential RP locations – are connected by roads that are represented by a set of directed arcs $\mathcal{A}$. Utilizing this network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, the construction of an RP-network in Model 1 involves the determination of 1) RP locations, 2) nonRP nodes assignment, and 3) the actual route for each commodity, in such a way that the total RP location cost and the total commodity transportation cost are minimized. Moreover, these decisions are made under the tour length/distance constraints in order to allow the local and lane truck drivers to relay shipments without traveling farther than Δ_1 and Δ_2 away from their home

base RPs. In addition to the tour length constraints, two other requirements are the load-imbalance and the percentage circuitry constraints. These two constraints help control the empty mileage and additional travel distance.

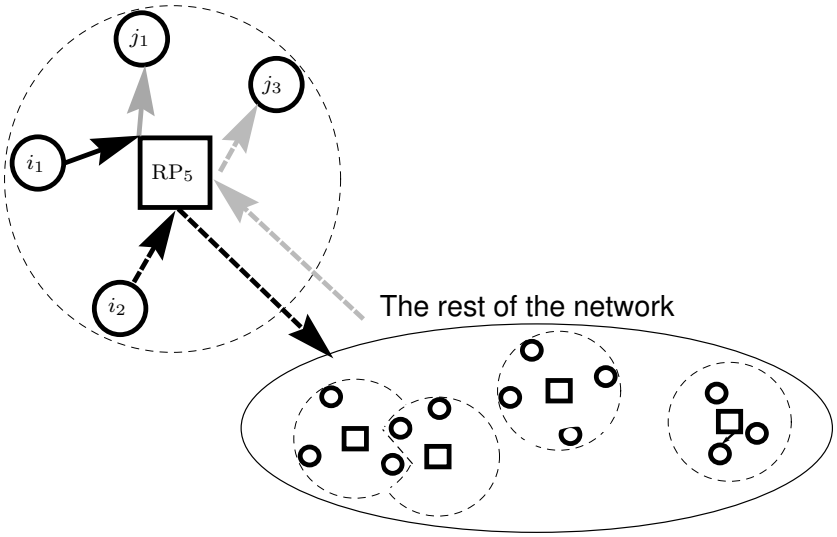
As illustrated in Figure 2, the RP and nonRP nodes are represented by squares and circles, respectively. Associated with each RP is the dashed contour line, whereby every nonRP node inside this contour is at most Δ_1 from and is assigned to the associated RP at the center. Moreover, the RP-induced network forms a connected network with respect to the distance Δ_2. In Figure 2, one example commodity $[i, j]$ is routed through RP_1, RP_2, RP_3, and RP_4. To comply with the percentage circuitry constraints, the total distance from i to j must not be more than Ω percent greater than the direct distance between i and j, d_{ij}. For the load-imbalance constraints that are defined for every located RP, we use RP_5 in Figure 3 to provide the illustration of this requirement.

Figure 2: A Schematic View of Model 1



In Figure 3, the solid arrow represents the commodities with both origin and destination in the same region. In this intra-region case, local drivers pick up and deliver the shipments; the pick-up is color coded black, whereas the delivery is grey. For the inter-region flow, the black dashed arrow represents the outgoing commodities that originate within the region of RP_5, but with a destination in another region. In contrast, the dashed grey arrow represents incoming commodities with a destination inside RP_5's region, but that originated elsewhere. The local drivers only pick up shipments in the former case and only deliver shipments in the latter case.

Figure 3: Load-Imbalance Constraints



Clearly, a large difference between the pick-ups and deliveries in a region, as caused by inter-region flows (intra-region flows do not provoke the load-imbalance level as drivers perform both pick-ups and deliveries), implies a high load-imbalance level, which directly leads to high level of empty mileage. In this case, we consider keeping the load-imbalance to a low level even though it does not ensure low empty

mileage. In doing this, the resulting RP-network will be composed of regions with a balanced number of pick-ups and deliveries that, if coupled with an efficient local routing procedure, can potentially help with empty mileage management. However, local routing decisions should be made at the operations level and are beyond the scope of our research, which focuses more on the strategic network-design level. Finally, we note that the load-imbalance and the percentage circuitry constraints are developed by Üster and Maheshwari (2007), who provide detailed derivation and discussion of them.

IV.1.1. Model Formulation

Utilizing the notation presented in Section II.1, the load-imbalance constraints and the percentage circuitry constraints can be stated as follows:

$$\sum_i \sum_j w_{ij}\, x_{ik} - \sum_i \sum_j w_{ij}\, x_{jk} \leq \Psi \sum_i \sum_j w_{ij}\, x_{ik} \qquad \forall\, k \in \mathcal{N} \qquad (4.1)$$

$$\sum_i \sum_j w_{ij}\, x_{jk} - \sum_i \sum_j w_{ij}\, x_{ik} \leq \Psi \sum_i \sum_j w_{ij}\, x_{jk} \qquad \forall\, k \in \mathcal{N} \qquad (4.2)$$

$$\Big(\sum_k d_{ik}\, x_{ik} + \sum_k \sum_l d_{kl}\, y_{kl}^{ij} + \sum_k d_{jk}\, x_{jk} \Big) - d_{ij} \leq \Omega\, d_{ij} \qquad \forall\, [i,j] \in \mathcal{Q} \qquad (4.3)$$

In constraints (4.1) and (4.2), the terms $\sum_i \sum_j w_{ij}\, x_{ik}$ and $\sum_i \sum_j w_{ij}\, x_{jk}$ correspond to the local drivers' pick-ups and deliveries. According to these two constraints, the difference between the pick-ups and deliveries cannot exceed Ψ percent of the the larger one. In constraints (4.3), the left hand side represents the additional travel distance of a commodity where the term in parentheses is the total distance when the commodity is routed through the RP-network. Similar to the previous two constraints, the additional distance cannot exceed Ω percent of the direct shipment distance, d_{ij}.

By incorporating constraints (4.1), (4.2), and (4.3) into the base model's formulation (2.1)-(2.9), the complete formulation of Model 1 is as follows:

$$\text{Min } Z = \sum_i \sum_k T_1 d_{ik} \sum_j (w_{ij} + w_{ji}) x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2 d_{kl}\, w_{ij}\, y_{kl}^{ij} + \sum_k F_k\, x_{kk} \tag{4.4}$$

subject to

$$d_{ik}\, x_{ik} \leq \Delta_1 \qquad \forall i, k \in \mathcal{N} \tag{4.5}$$

$$d_{kl}\, y_{kl}^{ij} \leq \Delta_2 \qquad \forall [i,j] \in \mathcal{Q}, \forall k, l \in \mathcal{N} \tag{4.6}$$

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = x_{jk} - x_{ik} \qquad \forall [i,j] \in \mathcal{Q}, \forall k \in \mathcal{N} \tag{4.7}$$

$$\sum_i \sum_j w_{ij}\, x_{ik} - \sum_i \sum_j w_{ij}\, x_{jk} \leq \Psi \sum_i \sum_j w_{ij}\, x_{ik} \qquad \forall k \in \mathcal{N} \tag{4.8}$$

$$\sum_i \sum_j w_{ij}\, x_{jk} - \sum_i \sum_j w_{ij}\, x_{ik} \leq \Psi \sum_i \sum_j w_{ij}\, x_{jk} \qquad \forall k \in \mathcal{N} \tag{4.9}$$

$$\sum_k d_{ik} x_{ik} + \sum_k \sum_l d_{kl} y_{kl}^{ij} + \sum_k d_{jk} x_{jk} - d_{ij} \leq \Omega d_{ij} \qquad \forall [i,j] \in \mathcal{Q} \tag{4.10}$$

$$\sum_k x_{ik} = 1 \qquad \forall i \in \mathcal{N} \tag{4.11}$$

$$x_{ik} \leq x_{kk} \qquad \forall i, k \in \mathcal{N} \tag{4.12}$$

$$y_{kl}^{ij} \leq x_{kk} \qquad \forall [i,j] \in \mathcal{Q}, \forall k, l \in \mathcal{N} \tag{4.13}$$

$$y_{kl}^{ij} \leq x_{ll} \qquad \forall [i,j] \in \mathcal{Q}, \forall k, l \in \mathcal{N} \tag{4.14}$$

$$x_{ik}, \in \{0,1\},\ 0 \leq y_{kl}^{ij} \leq 1 \qquad \forall\, i,\ j,\ k,\ l \in \mathcal{N} \tag{4.15}$$

We first note that this mathematical formulation is based on the model given by Üster and Maheshwari (2007); however, our model includes constraints (4.11) and

eliminates the following three constraints which are now redundant,

$$\sum_k \sum_l y_{ik}^{ij} \geq 1, \qquad y_{kk}^{ij} \leq x_{ik}, \qquad y_{kk}^{ij} \leq x_{jk} \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k \in \mathcal{N}. \tag{4.16}$$

Constraints (4.16), along with the first term of the objective function, ensure that all of the commodities utilize the RP-network, whereas constraints (4.11) handle this requirement more directly. In addition, based on our computational studies, the formulation with (4.11) provides tighter bounds when a Branch-and-cut (e.g., using CPLEX) method is used. We also note that the inclusion of the redundant constraints (4.16) provides no additional computational benefits.

IV.1.2. Benders Decomposition Framework

According to our early discussion, solving the entire formulation of Model 1 with commercial Branch-and-cut software is not an effective approach because of the rapid growth of problem size. However, when the values of $\mathbf{x}$ variables are given, the reduced problem that contains only $\mathbf{y}$ variables is a linear program (LP) and can be further decomposed for each commodity. Such LP and decomposable structures make Model 1 and the base model (and also Model 3 presented in Chapter V) amenable to solution by Benders decomposition (BD).

To develop a BD based algorithm for Model 1, we observe that if the $\mathbf{x}$ values are fixed to satisfy constraints (4.5), (4.8), (4.9), (4.11), and (4.12), then the resulting problem over $\mathbf{y}$ variables (second term in (4.4) along with constraints (4.7) and (4.14)) is an uncapacitated multicommodity network flow problem with a side constraint (4.6). Constraints (4.6) can be eliminated by assigning an arbitrarily large distance to each arc (u, v) with a d_{uv} value greater than Δ_2. This allows us to always obtain a solution to the resulting problem whose infeasibility, if it exists, is simply marked by an unrealistically large objective value. Moreover, the resulting problem is separable

into $|\mathcal{Q}|$ problems and it can be shown that each such problem is in fact a shortest path problem on the RP-network.

Based on these observations, in what follows, we first consider our problem without the circuitry constraints (4.10) and present a base BD algorithm, along with details on algorithmic enhancement that include generation of strengthened (Benders) cuts, cut disaggregation schemes, feasibility seeking (ε-optimal) framework, and the use of a local search heuristic for improving the upper bounds. Later, in Section IV.1.4, we generalize our ε-optimal BD algorithm so as to effectively handle the circuitry constraints.

IV.1.2.1. Base BD Framework

The BD technique involves decomposing an overall formulation into a master problem and a subproblem, and then solving them iteratively by utilizing the solution of the one in the other (Benders, 1962). The "subproblem" includes continuous variables and associated constraints and the "master problem" contains integer variables and one additional (auxiliary) continuous variable that relates the subproblem to the master problem. An optimum solution to the master problem gives a set of values for the integer variables, as well as a valid lower bound for the overall objective value. Using the fixed integer variable values as input, the solution to the "dual subproblem" is used to calculate an upper bound and to construct a Benders cut. This Benders cut is added to the master problem in the next iteration and the iterative process continues in this fashion by solving the master problem and the dual subproblem until it is terminated upon a predetermined small optimality gap between the upper bound and the lower bound. The addition of a Benders cut to the master problem tightens the lower bound, which monotonically increases in the course of iterations; it is well-known that an optimal solution is reached if enough iterations are completed.

IV.1.2.2. Benders Subproblem and its Dual

For given $\hat{\mathbf{x}}$ variables, we can state a subproblem SP($\mathbf{y}|\hat{\mathbf{x}}$) for our formulation as follows:

$$\text{Min} \qquad Z_{\text{SP}} = \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} \tag{4.17}$$

subject to

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = \hat{x}_{jk} - \hat{x}_{ik} \qquad \forall\, k \in \mathcal{N},\ \forall\, [i,j] \in \mathcal{Q} \tag{4.18}$$

$$y_{kl}^{ij} \le \hat{x}_{kk} \qquad \forall\, k,\, l \in \mathcal{N},\ \forall\, [i,j] \in \mathcal{Q} \tag{4.19}$$

$$y_{kl}^{ij} \le \hat{x}_{ll} \qquad \forall\, k,\, l \in \mathcal{N},\ \forall\, [i,j] \in \mathcal{Q} \tag{4.20}$$

$$y_{kl}^{ij} \ge 0 \qquad \forall\, k,\, l \in \mathcal{N},\ \forall\, [i,j] \in \mathcal{Q} \tag{4.21}$$

It is clear that the subproblem SP($\mathbf{y}|\hat{\mathbf{x}}$) can be separated into $|\mathcal{Q}|$ problems SP$_{ij}$($\mathbf{y}|\hat{\mathbf{x}}$), $\forall\, [i,j] \in \mathcal{Q}$. Then, defining α_k^{ij}, σ_{kl}^{ij} and τ_{kl}^{ij} as the dual variables associated with (4.18), (4.19) and (4.20), respectively, the dual subproblem DSP$_{ij}$($\boldsymbol{\alpha}$, $\boldsymbol{\sigma}$, $\boldsymbol{\tau}|\hat{\mathbf{x}}$) for $[i,j] \in \mathcal{Q}$ is obtained as

$$\text{Max} \qquad Z_{\text{DSP}_{ij}} = \sum_k (\hat{x}_{jk} - \hat{x}_{ik})\, \alpha_k^{ij} + \sum_k \sum_l (\hat{x}_{kk}\, \sigma_{kl}^{ij} + \hat{x}_{ll}\, \tau_{kl}^{ij}) \tag{4.22}$$

subject to

$$\alpha_l^{ij} - \alpha_k^{ij} + \sigma_{kl}^{ij} + \tau_{kl}^{ij} \le T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N},\ k \ne l \tag{4.23}$$

$$\sigma_{kl}^{ij},\, \tau_{kl}^{ij} \le 0, \quad \alpha_k^{ij} \text{ unrestricted} \qquad \forall\, k,\, l \in \mathcal{N},\ k \ne l \tag{4.24}$$

After solving the dual subproblem, the Benders cut can be generated using the values of dual variables $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\tau}}$, and an auxiliary continuous variable B as follows:

$$B \ge \sum_i \sum_j \Big(\sum_k (x_{jk} - x_{ik})\, \hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk}\, \hat{\sigma}_{kl}^{ij} + x_{ll}\, \hat{\tau}_{kl}^{ij}) \Big) \tag{4.25}$$

Note that, if the network G is connected, the subproblem always has a feasible solution and its dual is bounded.

IV.1.2.3. Benders Master Problem

Given the values of the dual variables $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, and $\hat{\boldsymbol{\tau}}$, which are used to form Benders cuts, the master problem MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for our problem becomes

$$\text{Min} \qquad Z_{\text{MP}} = \sum_i \sum_k T_1 \, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_k F_k \, x_{kk} + \mathit{SumBvars} \tag{4.26}$$

subject to

$$d_{ik} \, x_{ik} \leq \Delta_1 \qquad \forall \, i, \, k \in \mathcal{N} \tag{4.27}$$

$$\sum_i \sum_j w_{ij} \, x_{ik} - \sum_i \sum_j w_{ij} \, x_{jk} \leq \Psi \sum_i \sum_j w_{ij} \, x_{ik} \qquad \forall \, k \in \mathcal{N} \tag{4.28}$$

$$\sum_i \sum_j w_{ij} \, x_{jk} - \sum_i \sum_j w_{ij} \, x_{ik} \leq \Psi \sum_i \sum_j w_{ij} \, x_{jk} \qquad \forall \, k \in \mathcal{N} \tag{4.29}$$

$$\sum_k x_{ik} = 1 \qquad \forall \, i \in \mathcal{N} \tag{4.30}$$

$$x_{ik} \leq x_{kk} \qquad \forall \, i, \, k \in \mathcal{N} \tag{4.31}$$

(constraints for the set of *BCuts*) (4.32)

$$x_{ik} \in \{0, 1\}, \; \mathit{Bvars} \geq 0 \qquad \forall \, i, \, k \in \mathcal{N} \tag{4.33}$$

As discussed in detail in Section IV.1.3.2, we consider different types of Benders cuts obtained via disaggregation of (4.25). Thus, in (4.32), we represent the generated Benders cuts generically by *BCuts*. Moreover, we use *Bvars* to refer to the auxiliary continuous variables associated with *Bcuts*, and *SumBvars* to refer to the sum of *Bvars*. For a typical Benders algorithm, constraints (4.32) are the same as constraints (4.25) and both terms *Bvars* and *SumBvars* are equal to B.

Algorithm 1 Base BD Algorithm for Model 1

1: Initialize $UB = \infty$, $Bvars = 0$, $\hat{\boldsymbol{\alpha}} = \hat{\boldsymbol{\sigma}} = \hat{\boldsymbol{\tau}} = 0$ and $Iter = 0$; $MaxIter$;
2: Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP} and $\hat{\mathbf{x}}$. Set $LB = Z_{MP}$;
3: **while** $Iter \leq MaxIter$ **do**
4: Solve DSP($\boldsymbol{\alpha}$, $\boldsymbol{\sigma}$, $\boldsymbol{\tau}|\hat{\mathbf{x}}$) for Z_{DSP}, $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$;
5: $Iter = Iter + 1$;
6: **if** $Z_{\text{MP}} - SumBvars + Z_{\text{DSP}} < UB$ **then**
7: $UB = Z_{\text{MP}} - SumBvars + Z_{\text{DSP}}$; $\bar{\mathbf{x}} = \hat{\mathbf{x}}$;
8: **end if**
9: **if** $(UB$ - $LB)/\ LB \leq \varepsilon$ **then**
10: **break**;
11: **end if**
12: Generate *BCuts* with $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$ and incorporate them into MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\tau}}$);
13: Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP}, $\hat{\mathbf{x}}$, and $Bvars$. Set $LB = Z_{MP}$;
14: **if** $(UB$ - $LB)/\ LB \leq \varepsilon$ **then**
15: **break**;
16: **end if**
17: **end while**
18: Solve SP($\mathbf{y}|\bar{\mathbf{x}}$) to obtain $\bar{\mathbf{y}}$;
19: $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is the best solution upon termination.

We present the base BD algorithm in Algorithm 1, in which UB, LB, and $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ denote the best upper bound, the best lower bound, and the best feasible solution, respectively. The parameters $Iter$ and $MaxIter$ are specified to count the number of times that the master problem is solved and maximum allowed count value, respectively. The algorithm terminates either when $Iter$ is greater than $MaxIter$ or when the optimality gap, $((UB$ - $LB)/\ LB)$, is no greater than $\varepsilon \geq 0$. In each iteration, the optimality gap is checked twice; once after solving the dual subproblem and once after solving the master problem, so that the algorithm is terminated as soon as the incumbent solution (corresponding to UB) is within ε from the optimal solution. Also note that, in line 4, the dual subproblem is solved after it is separated

for each $[i, j] \in \mathcal{Q}$ to problems $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$, and Z_{DSP} denotes the sum of these individual optimum objective function values.

IV.1.3. Approaches for Accelerating the Base Algorithm

The base BD algorithm, as given above, is not a satisfactorily efficient approach for our problem. Thus, we explore various techniques that can potentially help accelerate the algorithm to provide solutions with low optimality gaps in shorter runtimes.

IV.1.3.1. Strengthening the Benders Cuts

The subproblems $\text{SP}(\mathbf{y}|\hat{\mathbf{x}})$ and $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$, after separation for each $[i, j] \in \mathcal{Q}$, are network flow problems which commonly possess degeneracy. This causes the dual subproblem to have multiple optimal solutions, each of which defines a different Benders cut. Thus, it is important to determine an optimal solution to the $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$ that provides a stronger Benders cut (in lines 4 and 12 of the BD algorithm in Algorithm 1). Magnanti and Wong (1981) define the strongness of a Benders cut for an optimization problem $\text{Min}_{y \in Y,\, z \in R}\{z : z \geq f(u) + y\, g(u),\, \forall\, u \in U\}$ as follows: The cut $z \geq f(u^1) + y\, g(u^1)$ dominates or is stronger than the cut $z \geq f(u) + y\, g(u)$ if $f(u^1) + y\, g(u^1) \geq f(u) + y\, g(u),\, \forall\, y \in Y$ with a strict inequality for at least one $y \in Y$.

As stated earlier, subproblem $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ specified for a commodity $[i, j] \in \mathcal{Q}$ is essentially a shortest path problem. This can be seen as follows. Once a solution to the $\text{MP}(\cdot)$ is obtained, we readily know the locations of the RPs and the assignment of the nonRP nodes to these RPs. Then, utilizing the set of RP nodes, $\mathcal{N}_{\text{RP}}$ only, we can generate an RP-induced complete network, G_{RP}, in which the arcs with distance longer than Δ_2 have arbitrarily large arc distances as mentioned before. For a given commodity $[i, j]$, letting $r(j)$ be the RP to which the destination node j is assigned and $r(i)$ be the RP to which the origin node i is assigned, $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ poses the problem

of finding its least cost route from $r(i)$ to $r(j)$, i.e., the shortest path on the G_{RP} with arc lengths calculated as $(T_2\, d_{kl}\, w_{ij})$, $\forall\, k,\, l \in \mathcal{N}_{\text{RP}}$. Then, $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ can alternatively be stated as

$$\text{Min} \qquad Z_{\text{ASP}_{ij}} = \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij}$$

subject to

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = \begin{cases} 1 & \text{if } k = r(j) \\ 0 & \text{if } k \neq r(i),\, r(j) \\ -1 & \text{if } k = r(i) \end{cases} \qquad \forall\, k \in \mathcal{N}_{\text{RP}} \tag{4.34}$$

$$y_{kl}^{ij} \geq 0 \qquad \forall\, k,\, l \in \mathcal{N}_{\text{RP}} \tag{4.35}$$

By defining $\tilde{\alpha}_k^{ij}$ as the dual variables corresponding to constraints (4.34), the dual subproblem $\text{DASP}_{ij}(\boldsymbol{\alpha}|\hat{\mathbf{x}})$ is obtained as

$$\text{Max} \qquad Z_{\text{DASP}_{ij}} = \tilde{\alpha}_{r(j)}^{ij} - \tilde{\alpha}_{r(i)}^{ij}$$

subject to

$$\tilde{\alpha}_l^{ij} - \tilde{\alpha}_k^{ij} \leq T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N}_{\text{RP}} \tag{4.36}$$

$$\tilde{\alpha}_k^{ij} \quad \text{unrestricted} \qquad \forall\, k \in \mathcal{N}_{\text{RP}} \tag{4.37}$$

The above shortest path problem $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ can easily be solved using Dijkstra's algorithm. Letting the optimal shortest path distance from $r(i)$ to $r(j)$ be L_{ij}, it is clear that an optimal solution $\tilde{\boldsymbol{\alpha}}^*$ to $\text{DASP}_{ij}(\tilde{\boldsymbol{\alpha}}|\hat{\mathbf{x}})$ has $\tilde{\alpha}_{r(j)}^{ij^*}$ and $\tilde{\alpha}_{r(i)}^{ij^*}$ values as L_{ij} and 0, respectively. Moreover, an optimal solution $(\boldsymbol{\alpha}^*, \boldsymbol{\sigma}^*, \boldsymbol{\tau}^*)$ to $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$ is given by $(\tilde{\boldsymbol{\alpha}}^*, \mathbf{0}, \mathbf{0})$. Letting $\mathcal{A}_i$ be the set of nodes that are within Δ_1 distance of node i, i.e., $\mathcal{A}_i = \{k \in \mathcal{N} : d_{ik} \leq \Delta_1\}$, we observe, in the first sum of the objective

function of $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$, that a term for a node k is not ensured to be nullified only if either $k \in \mathcal{A}_i$ or $k \in \mathcal{A}_j$. Then, solving the following linear program provides stronger Benders cuts in the sense of strongness defined above.

$$\text{Max} \qquad \sum_{k \in \mathcal{A}_j} \alpha_k^{ij} - \sum_{k \in \mathcal{A}_i} \alpha_k^{ij} + \sum_k \sum_l (\sigma_{kl}^{ij} + \tau_{kl}^{ij}) \tag{4.38}$$

subject to

$$\alpha_{r(j)}^{ij} = L_{ij}, \qquad \alpha_{r(i)}^{ij} = 0 \tag{4.39}$$

$$\sum_k (\hat{x}_{jk} - \hat{x}_{ik}) \alpha_k^{ij} + \sum_k \sum_l (\hat{x}_{kk}\, \sigma_{kl}^{ij} + \hat{x}_{ll}\, \tau_{kl}^{ij}) = L_{ij} \tag{4.40}$$

$$\alpha_l^{ij} - \alpha_k^{ij} + \sigma_{kl}^{ij} + \tau_{kl}^{ij} \leq T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N},\ k \neq l \tag{4.41}$$

$$\sigma_{kl}^{ij},\, \tau_{kl}^{ij} \leq 0, \quad \alpha_k^{ij} \text{ unrestricted} \qquad \forall\, k,\, l \in \mathcal{N},\ k \neq l \tag{4.42}$$

Constraints (4.38) fix only the values of two variables in an optimal solution to dual subproblem $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$. Constraints (4.39), (4.40), and (4.41) ensure that the solution to the above problem is feasible for the $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}})$ and the implied Benders cut is valid.

IV.1.3.2. Cut Disaggregation Schemes

Since the subproblem is separable into $|\mathcal{Q}|$ independent shortest path problems, one for each commodity, this enables us to generate different types of Benders cuts. Specifically, we consider four alternative Benders cuts as follows:

Type A1 The first type of cut is the typical aggregate Benders cut. In each iteration, a single cut is included in the master problem. Then, we have $Bvars = B$,

$SumBvars = B$, and the cut is

$$B \geq \sum_i \sum_j \Big(\sum_k (x_{jk} - x_{ik})\, \hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk}\, \hat{\sigma}_{kl}^{ij} + x_{ll}\, \hat{\tau}_{kl}^{ij}) \Big)$$

Type D2 We disaggregate the Benders cut so that one cut is added for each node $i \in \mathcal{N}$ in which at least one commodity originates. Then, $Bvars = B_i,\ \forall\, i \in \mathcal{N}$, $SumBvars = \sum_i B_i$, and the cuts are

$$B_i \geq \sum_j \Big(\sum_k (x_{jk} - x_{ik}) \hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk} \hat{\sigma}_{kl}^{ij} + x_{ll} \hat{\tau}_{kl}^{ij}) \Big) \qquad \forall\, i \in \mathcal{N}$$

Type D3 We disaggregate the Benders cut so that one cut is added for each $j \in \mathcal{N}$ to which at least one commodity is destined. Then, $Bvars = B_j,\ \forall\, j \in \mathcal{N}$, $SumBvars = \sum_j B_j$, and the cuts are

$$B_j \geq \sum_i \Big(\sum_k (x_{jk} - x_{ik}) \hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk} \hat{\sigma}_{kl}^{ij} + x_{ll} \hat{\tau}_{kl}^{ij}) \Big) \qquad \forall\, j \in \mathcal{N}$$

Type D4 We disaggregate the Benders cuts so that one cut is added for each commodity $[i, j] \in \mathcal{Q}$. Therefore, we have $|\mathcal{Q}|$ cuts and $Bvars = B_{ij},\ \forall\, [i, j] \in \mathcal{Q}$, $SumBvars = \sum_{ij} B_{ij}$, and the cuts are given by

$$B_{ij} \geq \sum_k (x_{jk} - x_{ik}) \hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk} \hat{\sigma}_{kl}^{ij} + x_{ll} \hat{\tau}_{kl}^{ij}) \qquad \forall\, [i, j] \in \mathcal{Q}$$

In most cases, the use of multiple cuts can provide a tighter bound (Birge and Louveaux., 1998; Üster et al., 2007); however, the size and solution time of the master problem can increase dramatically depending on the type of Benders cuts employed. Typically, we expect an increasing runtime as we move from Type A1 to Type D4 cuts as given above.

IV.1.3.3. ε-Optimal Approach

Another approach to decreasing the excessive MP($\cdot$) runtime, whether multiple cuts are employed or not, is through the utilization of the ε-optimal approach introduced in Geoffrion and Graves (1974). In this approach, the MP($\cdot$) includes one additional constraint, given as

$$\sum_i \sum_k T_1 \, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_k F_k \, x_{kk} + \mathit{SumBvars} \leq \; UB(1-\varepsilon) \qquad (4.43)$$

where ε denotes the acceptable optimality gap. In an iteration, instead of solving the MP($\cdot$) to optimality, it is only verified that there exists a feasible solution with an objective function value less than or equal to $UB(1-\varepsilon)$. This is simply achieved by stopping the Branch-and-cut as soon as a feasible solution is obtained. The values of the $\mathbf{x}$ variables given by this feasible solution are then used to solve the subproblem and generate valid Benders cuts. A considerable amount of runtime can be saved since the MP($\cdot$) is not optimized; however, the feasible solution obtained is no longer a valid lower bound. Thus, in the ε-Optimal BD algorithm given in Algorithm 2, the optimality tests, on lines 9-11 and 14-16 of the base BD algorithm, are removed, and the algorithm terminates when the MP($\cdot$) cannot find a feasible solution, which verifies that the best incumbent solution is within ε from optimality.

Algorithm 2 ε-Optimal BD Algorithm

1: Initialize $UB = \infty$, $Bvars = 0$, $\hat{\boldsymbol{\alpha}} = \hat{\boldsymbol{\sigma}} = \hat{\boldsymbol{\tau}} = 0$ and $Iter = 0$;
2: Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP} and $\hat{\mathbf{x}}$;
3: **while** MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) has a feasible solution **do**
4: Solve DSP($\boldsymbol{\alpha}$, $\boldsymbol{\sigma}$, $\boldsymbol{\tau}|\hat{\mathbf{x}}$) for Z_{DSP}, $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$;
5: $Iter = Iter + 1$;
6: **if** $Z_{\text{MP}} - SumBvars + Z_{\text{DSP}} < UB$ **then**
7: $UB = Z_{\text{MP}} - SumBvars + Z_{\text{DSP}}$; $\quad \bar{\mathbf{x}} = \hat{\mathbf{x}}$;
8: Update the incumbent value UB in constraint (4.43);
9: **end if**
10: Generate *BCuts* with $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$ and incorporate them into MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\tau}}$);
11: Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP}, $\hat{\mathbf{x}}$, and $Bvars$;
12: **end while**
13: Solve SP($\mathbf{y}|\bar{\mathbf{x}}$) to obtain $\bar{\mathbf{y}}$;
14: $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is the best solution upon termination.

IV.1.3.4. A Heuristic Algorithm to Enhance the Upper Bound

In both the Base and the ε-optimal BD algorithms, upper bounds can be improved using heuristics so that improved optimality gaps and, thus, a faster convergence, are achieved. For example, in the ε-optimal approach, it is clear that the constraint (4.43) becomes stronger as the value of UB decreases. In fact, this constraint is relatively weak in early iterations since the value of UB is usually large.

Our heuristic local search algorithm is aimed at quickly conducting a neighborhood search for an improved UB, and it employs the most recent MP($\cdot$) solution $\hat{\mathbf{x}}$ obtained as its initial solution. In particular, we represent a solution in the heuristic algorithm by a set of opened RPs $\mathcal{S} \subseteq \mathcal{N}$, which is given by the nodes $i \in \mathcal{N}$ whose $\hat{x}_{ii}$ value is one in the MP($\cdot$) solution. The assignments of nonRP nodes ($\mathcal{N} \setminus \mathcal{S}$) to the RPs are initially given again by the solution $\hat{\mathbf{x}}$. Observe that $\hat{\mathbf{x}}$ satisfies the imbalance constraints (4.8) and (4.9), which are included in the MP($\cdot$). Given the RP

locations, i.e., a solution $\mathcal{S}$, and a feasible assignment of nonRP nodes to RPs (with respect to imbalance constraints), a complete solution can be obtained by determining the actual shortest path for each commodity on the RP-network. The goodness of a neighboring solution $\mathcal{S}$, $Z(\mathcal{S})$, can then be evaluated using the objective function (4.4).

In the heuristic algorithm, outlined in Algorithm 3, we employ three types of neighborhood functions, represented by the sets *Add*, *Drop*, and *Swap*. *Add* is the set of nodes that are currently nonRP, thus, they are candidates for being added as RPs. *Drop* includes the current RP nodes (i.e., the nodes in $\mathcal{S}$) which can be made nonRP. *Swap* involves the node-pairs (u, v) where u is an RP and v is a nonRP, thus, their roles can be swapped. In each iteration, neighboring solutions of the current solution $\mathcal{S}$ are generated and examined using each of the neighborhoods.

More specifically, first, in examining the *Add* neighborhood (lines 5-11), we randomly pick a node u from *Add*, form $\mathcal{S}_{\text{nhbd}}$ and update *Add*. Then, we reassign the new RP to itself and unassign it from the RP to which it was previously assigned as a nonRP. For simplicity and runtime considerations, we do not reroute the TLs through the new set of RPs from scratch. Thus, the reassignment affects the imbalance constraint (which we recalculate) only at the latter node since the new node used to be in its region. Notice that the imbalance constraints only include $\mathbf{x}$ variables and the new RP automatically satisfies the imbalance constraint. $\mathcal{S}_{\text{nhbd}}$ is recorded as $\mathcal{S}_{\text{temp}}$ if it is feasible and has an improved objective value.

Second, we examine the *Drop* neighborhood (lines 12-18) similarly by randomly picking an RP v from *Drop*. However, when an RP is excluded from $\mathcal{S}$, it is possible that the RP-induced network is now disconnected, leading to an infeasibility. On the other hand, the nodes in RP v's region need to be re-assigned to the RPs that are in $\mathcal{S}_{\text{nhbd}}$ so that the imbalance constraints are satisfied. For this, we randomly select

now-unassigned nodes one at a time and assign them to the existing RPs greedily in such a way that, at each step, the implied new value of load-imbalance in the network is minimized. Again, $\mathcal{S}_{\text{nhbd}}$ is recorded as $\mathcal{S}_{\text{temp}}$ if it provides a feasible solution with improved value.

Thirdly, we consider the *Swap* neighborhood (lines 19-25) again by randomly picking pairs of RP and nonRP nodes. In this case, we first add the nonRP node u (as in the *Add* neighborhood above but without checking for imbalance) and then proceed with dropping the node v exactly as in the *Drop* neighborhood above. Finally, if the newly obtained $\mathcal{S}_{\text{nhbd}}$ has an improved objective value and is feasible we record it as $\mathcal{S}_{\text{temp}}$. The search of three neighborhoods (lines 4-26) continues until the best solution of the three (recorded as $\mathcal{S}_{\text{temp}}$) improves the incumbent or the sets *Add*, *Drop*, and *Swap* are non-empty. The overall procedure (while loop) is continued until no improving solution is found.

Algorithm 3 Heuristic Algorithm for improving the Upper Bound

$\mathcal{S}_{\text{temp}} = \mathcal{S}$
while $\mathcal{S} = \mathcal{S}_{\text{temp}}$ **do**
  $Add = \mathcal{N} \backslash \mathcal{S}$ $Drop = \mathcal{S}$; $Swap = \{(u, v) : u \in \mathcal{N} \backslash \mathcal{S}, v \in \mathcal{S}\}$
  repeat
    if *Add* is non-empty **then**
      Randomly pick a node $u \in Add$
      $\mathcal{S}_{\text{nhbd}} = \mathcal{S} \cup \{u\}$ $Add = Add \setminus \{u\}$
      if $\mathcal{S}_{\text{nhbd}}$ is feasible and $Z(\mathcal{S}_{\text{nhbd}}) < Z(\mathcal{S}_{\text{temp}})$ **then**
        $\mathcal{S}_{\text{temp}} = \mathcal{S}_{\text{nhbd}}$
      end if
    end if
    if *Drop* is non-empty **then**
      Randomly pick a node $v \in Drop$
      $\mathcal{S}_{\text{nhbd}} = \mathcal{S} \setminus \{v\}$ $Drop = Drop \setminus \{v\}$
      if $\mathcal{S}_{\text{nhbd}}$ is feasible and $Z(\mathcal{S}_{\text{nhbd}}) < Z(\mathcal{S}_{\text{temp}})$ **then**
        $\mathcal{S}_{\text{temp}} = \mathcal{S}_{\text{nhbd}}$
      end if
    end if
    if *Swap* is non-empty **then**
      Randomly pick a node pair $(u, v) \in Swap$
      $\mathcal{S}_{\text{nhbd}} = \mathcal{S} \cup \{u\} \setminus \{v\}$ $Swap = Swap \setminus \{(u, v)\}$
      if $\mathcal{S}_{\text{nhbd}}$ is feasible and $Z(\mathcal{S}_{\text{nhbd}}) < Z(\mathcal{S}_{\text{temp}})$ **then**
        $\mathcal{S}_{\text{temp}} = \mathcal{S}_{\text{nhbd}}$
      end if
    end if
  until $\mathcal{S}_{\text{temp}} \neq \mathcal{S}$ or $Add = Drop = Swap = \emptyset$
  if $Z(\mathcal{S}_{\text{temp}}) < Z(\mathcal{S})$ **then**
    $\mathcal{S} = \mathcal{S}_{\text{temp}}$
  else
    break;
  end if
end while

To obtain heuristic enhanced Base and ε-Optimal BD algorithms, we modify the pseudocodes in Algorithms 1 and 2 in the same way. Let Z_{Heur} and $\hat{\mathbf{x}}_{\text{Heur}}$ denote the objective value and the values of the $\mathbf{x}$ after the heuristic is applied. Then, specifically,

we include an additional step just after line 5 to state "Apply the Heuristic Algorithm initiated with $\hat{\mathbf{x}}$," and also modify lines 6 and 7 as " **if** $Z_{\text{Heur}} \leq UB$ **then**" and "$UB = Z_{\text{Heur}}$; $\bar{\mathbf{x}} = \hat{\mathbf{x}}_{\text{Heur}}$," respectively. Notice that, once the DSP$(\cdot)$ is solved on line 4 of this algorithm, the objective value of the initial solution of the heuristic is obtained. Thus, the heuristic should provide an upper bound (Z_{Heur}) that is no worse than the initial solution, which was used as an upper bound before the heuristic was employed.

IV.1.4. Including the Percentage Circuitry Constraints

We incorporate the percentage circuitry constraints (4.10) into the formulation and the Benders algorithm. Since these constraints contain $\mathbf{y}$ variables, they are included in the subproblem (4.17) - (4.21). By defining η^{ij} as associated dual variables, the dual subproblem DSP$_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}, \boldsymbol{\eta}|\hat{\mathbf{x}})$ for $[i,j] \in \mathcal{Q}$ becomes

$$\begin{aligned} \text{Max} \qquad Z_{\text{DSP}_{ij}} = & \sum_k (\hat{x}_{jk} - \hat{x}_{ik})\, \alpha_k^{ij} + \sum_k \sum_l (\hat{x}_{kk}\, \sigma_{kl}^{ij} + \hat{x}_{ll}\, \tau_{kl}^{ij}) \\ & + \Big((\Omega+1) d_{ij} - \sum_k d_{ik}\, \hat{x}_{ik} - \sum_k d_{jk}\, \hat{x}_{jk} \Big) \eta^{ij} \end{aligned} \tag{4.44}$$

subject to

$$\alpha_l^{ij} - \alpha_k^{ij} + \sigma_{kl}^{ij} + \tau_{kl}^{ij} + d_{kl}\, \eta^{ij} \leq T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N},\ k \neq l \tag{4.45}$$

$$\sigma_{kl}^{ij},\, \tau_{kl}^{ij},\, \eta^{ij} \leq 0, \quad \alpha_k^{ij} \text{ unrestricted} \qquad \forall\, k,\, l \in \mathcal{N},\ k \neq l \tag{4.46}$$

For a commodity $[i,j] \in \mathcal{Q}$, if the SP$_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ is infeasible, which can only be due to circuitry constraints, then the solution to the DSP$_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}, \boldsymbol{\eta}|\hat{\mathbf{x}})$ is unbounded. In this case, we add the following Benders cut (4.47) which is based on the extreme ray

to the MP($\cdot$).

$$\sum_k (x_{jk} - x_{ik})\hat{\alpha}_k^{ij} + \sum_k \sum_l (x_{kk}\hat{\sigma}_{kl}^{ij} + x_{ll}\hat{\tau}_{kl}^{ij}) + \Big((\Omega+1)d_{ij} - \sum_k d_{ik}x_{ik} - \sum_k d_{jk}x_{jk}\Big)\hat{\eta}^{ij} \leq 0 \tag{4.47}$$

The infeasibility can easily be checked by verifying the validity of the inequality $L_{ij} \leq T_2\, w_{ij}\left((\Omega+1)d_{ij} - d_{ir(i)} - d_{jr(j)}\right)$, $[i,j] \in \mathcal{Q}$. If this inequality holds, i.e., the $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}})$ is feasible, then the circuitry constraint can be discarded (or set η^{ij} to zero) and a strengthened Benders cut is generated exactly as before by solving the problem (4.38)-(4.42) given in Section IV.1.3.1. Although these cuts, one for each commodity $[i,j] \in \mathcal{Q}$, can be aggregated as with the different cut types presented in Section IV.1.5, we employ the Type D4 cuts in our computational studies. We also note that, in the presence of circuitry constraints, when the local search heuristic is used to improve the *UB* as described above, we calculate the shortest path L_{ij} on the RP-network implied by a neighboring solution $\mathcal{S}_{\text{nhbd}}$ and discard the solution if it is infeasible.

IV.1.4.1. Derivation of Surrogate Constraints

The convergence rate of the ε-Optimal BD algorithm with circuitry constraints included can be slow if the master problem provides an underlying network that does not encourage feasibility for the subproblem, which leads to the addition of Benders cuts mostly in the form of extreme rays. In our computational studies, we observed that this indeed is the case and the infeasibility (with respect to circuitry constraints) is primarily caused by the expression $\left((\Omega+1)d_{ij} - \sum_k d_{ik}\hat{x}_{ik} - \sum_k d_{jk}\hat{x}_{jk}\right)$ having a negative value for a number of commodities. This kind of infeasibility can be reduced

by including the following surrogate constraints (4.48) into the MP(·).

$$(\Omega+1)d_{ij} - \sum_k d_{ik}\, x_{ik} - \sum_k d_{jk}\, x_{jk} \geq 0 \qquad \forall\, [i,j] \in \mathcal{Q}. \tag{4.48}$$

However, a tighter surrogate constraint set can be derived from the percentage circuitry constraints. For this purpose, we first rewrite the constraints (4.10) as

$$(\Omega+1)d_{ij} - \sum_k d_{ik}\, x_{ik} - \sum_k d_{jk}\, x_{jk} \geq \sum_k \sum_l d_{kl}\, y_{kl}^{ij} \qquad \forall\, [i,j] \in \mathcal{Q}. \tag{4.49}$$

In the optimal solution to the overall problem, the values of the terms B_{ij} and $\sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij}$ must be equal in the MP(·) for each commodity $[i,j] \in \mathcal{Q}$ (recall that the subproblem is separable). Then, the right-hand side of the (4.49) can be replaced accordingly and the following surrogate constraints are then included in the MP(·).

$$\Big((\Omega+1)d_{ij} - \sum_k d_{ik}\, x_{ik} - \sum_k d_{jk}\, x_{jk}\Big) \geq B_{ij}/(T_2\, w_{ij}) \qquad \forall\, [i,j] \in \mathcal{Q}. \tag{4.50}$$

Although not a guarantee, adding the constraints (4.50) to the MP(·) encourages MP(·) to provide a solution $\hat{\mathbf{x}}$ that facilitates feasibility in the subproblems.

IV.1.5. Computational Experiments

We conduct computational experiments to evaluate and compare the performance of our BD algorithms. The comparisons illustrate the benefit of utilizing strengthened Benders cuts, cut disaggregation schemes, a heuristic to enhance upper bounds, ε-optimal framework, and the surrogate constraints which are employed when circuitry constraints are included. In addition, our computational study also helps us to examine the influence of problem parameters on the performance of the algorithms. We also note that, in order to solve the master problem, the dual subproblem, and the original problem with a Branch-and-cut approach, we use CPLEX 9.1 with default

settings for cut generation, preprocessing, and upper bound heuristics. As discussed earlier, all the experiments are conducted using C++ with STL (Standard Template Library) whenever possible and Concert Technology (ILOG, Inc.).

In the first experiment, we examine the performance of a Branch-and-cut approach (as implemented in CPLEX) and a variety of Benders decomposition algorithms. In Table 7, we present the average runtimes for solving four problem classes, Ua1-Ud1 where each class consists of 10 random instances with varying $|\mathcal{N}|$ values as shown and a D value of 20. As observed in the third and fourth columns, the inclusion of the constraints (4.11) promotes faster solution times; however, for instances with more than 40 nodes (corresponding to 320 commodities in this case), the memory requirements grow prohibitively large; thus, no solutions could be obtained. The fifth column includes the average runtime for the base BD algorithm, as presented in Section IV.1.2.1, which appears highly inefficient. The average runtime for C1 is only over four instances which are solved in a preset time limit of 5000 seconds (due to excessive runtimes with very high optimality gaps) and none of the other instances could be solved in that time frame.

As illustrated in the last two columns (with stopping criteria of 0.0% and 2.0% optimality gaps, respectively), the runtime of the BD algorithm can be significantly reduced by employing stronger cuts. This also enables us to solve larger instances that CPLEX or the base BD algorithm cannot solve in reasonable time frames; however, it is clear that a considerable runtime is required to close the last 2.0% optimality gap. Therefore, in the following experiments, we employ the base BD algorithm with strong cuts and set the stopping criterion to 2.0% optimality gap, including the ε in the ε-optimal BD approach.

In the second experiment, we compare the performance with varying cut disaggregation schemes in the context of both the Base and ε-optimal BD algorithms.

Table 7: Average runtimes (secs.) for BC and BD approaches

Problem Class	$\|\mathcal{N}\|$–D	Branch-and-Cut without (4.11)	Branch-and-Cut with (4.11)	Base BD Cut A1	BD–strengthened cuts A1 Optimal	BD–strengthened cuts A1 2.0% gap
Ua1	20–20	102.3	3.8	976.78	6.7	5.5
Ub1	25–20	1320.0	26.4	>5000	24.7	16.0
Uc1	30–20	2350.0	99.7	>5000	121.8	45.6
Ud1	40–20	n/s	n/s	>5000	1417.7	213.7

We employ six larger problem classes (Ud1-2, Ue1-2, and Uf1-2), each with 10 instances, and the average runtime results in Table 8 clearly indicate that the use of disaggregated cuts, especially the Type D4 cut, provides significantly better performance compared to the use of a single Benders cut (Type A1). Thus, in our following experiments, we employ Type D4 cuts in the BD algorithms building on Base and ε-optimal approaches.

Table 8: Average runtimes (secs.) with alternative Benders cuts

Base BD algorithm					
Class	$\|\mathcal{N}\|$–D	A1	D2	D3	D4
Ud1	40–20	209	55	51	43
Ud2	40–40	289	84	75	59
Ue1	60–20	2442	414	371	297
Ue2	60–40	1586	555	449	346
Uf1	80–20	10540	2053	1408	1441
Uf2	80–40	7696	2098	1942	1382
ε-optimal BD algorithm					
Class	$\|\mathcal{N}\|$–D	A1	D2	D3	D4
Ud1	40–20	263	56	56	38
Ud2	40–40	277	99	84	65
Ue1	60–20	2671	456	430	297
Ue2	60–40	1548	594	465	349
Uf1	80–20	10939	1841	1529	983
Uf2	80–40	7280	2059	1926	1347

In the third experiment, we examine the impact of using the heuristic algorithm (Section IV.1.3.4) to improve the *UB* in the Base and ε-optimal BD approaches. For this experiment, we utilized the problem class Uf1 with different settings of Δ_1–Δ_2 with 10 instances in each setting. The results are given in Table 9 in which the last column indicates the percentage decrease in average runtime when the heuristic algorithm is employed to improve the upper bounds.

Table 9: Results for BD approaches with and without upper bound heuristics

	Base BD Algorithm								
	No upper bound heuristic				Heuristic enhanced				Time
	Ave	Ave	Ave time/iter.		Ave	Ave	Ave time/iter.		red.
Δ_1–Δ_2	Time	*Iter*	MP	SP	Time	*Iter*	MP	SP	(%)
20–40	1441	4.7	111.5	192.0	1076	4.2	56.9	190.2	25.3
20–50	534	3.0	4.8	173.5	480	2.6	4.8	175.9	10.2
30–50	1011	3.6	67.3	210.2	883	3.5	38.2	211.6	12.5
30–60	612	3.0	12.4	192.2	598	2.9	10.9	193.0	2.4
	ε-optimal BD Algorithm								
	No upper bound heuristic				Heuristic enhanced				Time
	Ave	Ave	Ave time/iter.		Ave	Ave	Ave time/iter.		red.
Δ_1–Δ_2	Time	*Iter*	MP	SP	Time	*Iter*	MP	SP	(%)
20–40	983	4.6	20.6	190.3	876	4.1	16.3	189.9	10.9
20–50	559	3.2	1.9	172.1	486	2.7	1.2	174.4	13.2
30–50	1015	4.5	16.0	208.4	853	3.8	18.5	202.7	16.0
30–60	611	3.1	3.3	192.8	570	3.0	4.0	183.3	6.7

Notably, in the case of ε-optimal BD approach, the improved upper bound is utilized in constraint (4.43), which is added to the master problem, whereas in the base BD approach, the heuristic solution only affects the optimality gap calculations. Thus, the percent reduction in runtimes is more pronounced in the ε-optimal approach, although it is still quite significant in the base BD algorithm. In addition, the number of iterations decreases in both cases, but more significantly in the ε-optimal approach. Interestingly, since the subproblem solutions that generate stronger cuts

consume more computational time, any reductions in the number of iterations essentially improves overall runtime due to the reduced number of subproblems solved. Having observed its benefits, in the following experiments, we also incorporate the heuristic algorithm into both Base and ε-optimal BD approaches.

In the fourth experiment (see Table 10), we compare the performance of the Base algorithm and ε-optimal approaches: in both algorithms, we employ upper bound heuristic and the strengthened cuts of Type D4. We utilize the problem classes Uf1 and Uf2 ($|\mathcal{N}| = 80$, $D = 20$ and 40) with different settings of Δ_1–Δ_2 (20–40, 20–50, 30–50, and 30–60), and Ψ (1.0 and 0.2) – again with 10 instances for each setting.

Table 10: Results for BD approaches with varying Δ_1, Δ_2 and Ψ values

			$\Psi = 1.0$				$\Psi = 0.2$			
Class	Δ_1–Δ_2	Algorithm	Ave Time	Time red. %	Ave *Iter*	Ave MP time	Ave Time	Time red. %	Ave *Iter*	Ave MP time
Uf1	20–40	Base	1091		4.3	255	1242		4.1	426
		ε-opt	867	20.5	4.1	65	927	25.1	4.5	65
	20–50	Base	480		2.6	13	505		2.8	7
		ε-opt	478	0.5	2.7	4	492	2.5	2.8	5
	30–50	Base	888		3.5	133	1270		3.8	450
		ε-opt	856	3.6	3.9	61	950	25.2	4.2	84
	30–60	Base	598		2.9	31	648		3.0	57
		ε-opt	561	6.1	2.9	12	660	-2.0	3.5	13
Uf2	20–40	Base	1315		3.1	48	1398		3.2	113
		ε-opt	1267	3.6	3.1	8	1294	7.4	3.2	11
	20–50	Base	817		2.1	8	881		2.3	14
		ε-opt	809	0.9	2.1	3	951	-8.0	2.5	5
	30–50	Base	1778		3.1	362	2256		3.1	858
		ε-opt	1374	22.7	3.1	9	1663	26.3	3.2	252
	30–60	Base	1384		2.7	234	1970		2.8	791
		ε-opt	1134	18.1	2.7	18	1589	19.4	3.1	332

Inspecting columns 5 and 9, the results largely show that, with a few exceptions, employing the ε-optimality framework is beneficial to improving the solution times

over the base BD approach, even though it does not significantly decrease the number of iterations. When the value of Ψ reduces from 1.0 to 0.2, the problem becomes more difficult, especially for the master problem (which includes the imbalance constraint and is solved using Branch-and-cut). In turn, the solution times are longer - observed by comparing columns 4 and 8. In terms of tour lengths, recall that an increase in Δ_2 value provides an underlying network with higher connectivity, i.e., more arcs without artificially large arc lengths become available in the subproblem. This, in turn, helps to find better upper bounds and decrease the number of iterations, which lead to reduced runtimes for both master and subproblems, and, thus, a reduced total runtime. This can easily be observed by inspecting corresponding values for changing Δ_1–Δ_2 settings, i.e., 20-40 vs. 20–50 and 30–50 vs. 30–60, in both problem classes. Moreover, increasing the value of Δ_1 enlarges the solution space of the master problem. This reduces the lower bound quality provided by the master problem, which provides lower objective values with increased Δ_1. Specifically, comparing the results for Δ_1–Δ_2 values with 20–50 vs. 30–50 in each class, we observe that the number of iterations and the master problem runtimes increase – leading to increased total runtimes.

In the fifth experiment, we introduce the percentage circuitry into consideration (Section IV.1.4). We solve the instances in classes Uf1 and Uf2 under three Ψ–Ω combinations including 1.0–3.0, 1.0–2.0, and 0.3–3.0. The first two combinations correspond to having only the circuitry constraints effective and the last one effectively forces both imbalance and circuitry constraints. The results are reported in Table 11.

Table 11: Results for the ε-optimal BD approach with varying Δ_1–Δ_2 and Ψ–Ω values

Class	Δ_1–Δ_2	Ψ–Ω								
		1.0–2.0			1.0–3.0			0.3–3.0		
		Ave Time	$\bar{\Psi}^M$	$\bar{\Omega}^M$	Ave Time	$\bar{\Psi}^M$	$\bar{\Omega}^M$	Ave Time	$\bar{\Psi}^M$	$\bar{\Omega}^M$
Uf1	20-40	1880	0.41	1.94	955	0.37	2.87	**2199**[4]	0.26	2.84
	20-50	1569	0.40	1.91	629	0.33	2.79	670	0.27	2.73
	30-50	**8171**[1]	0.34	1.96	**2518**[3]	0.31	2.93	1653	0.25	2.86
	30-60	2294	0.33	1.96	1119	0.35	2.88	1090	0.26	2.83
Uf2	20-40	1991	0.34	1.94	1390	0.31	2.96	***1423***[5]	0.26	2.89
	20-50	2313	0.34	1.98	1174	0.34	2.87	***1148***[5]	0.26	2.94
	30-50	4019	0.31	1.99	2897	0.30	2.95	3290	0.26	2.99
	30-60	**4768**[2]	0.28	1.99	2689	0.34	2.92	3044	0.23	2.98

[1] The average of 9 instances without the outlier is 3916 seconds.
[2] The average of 9 instances without the outlier is 2982 seconds.
[3] The average of 9 instances without the outlier is 1627 seconds.
[4] The average of 9 instances without the outlier is 1063 seconds.
[5] Ψ is set to 0.35.

In Table 11, the entries in bold indicate the existence of outlier instances in terms of solution time. Specifically, a bold Ave Time value represents the average of solution times over all 10 instances – including the outlier instance. Furthermore, the entries in italics indicate that one of the instances is infeasible when the value of Ψ is 0.3. To solve this particular instance, we use a slightly increased Ψ value of 0.35 (instead of 0.30).

$\bar{\Psi}^M$ and $\bar{\Omega}^M$ columns represent the maximum level of the load-imbalance and the percentage circuitry over all 10 instances in a Δ_1–Δ_2 setting. These values were calculated after solving the instances with corresponding Ψ and Ω values. Interestingly, when the Ψ value is set to 1.0, which effectively eliminates the imbalance requirements, the $\bar{\Psi}^M$ ranges between 0.28 and 0.41, largely within the [0.30, 0.40] interval in the final solution. Thus, for active imbalance constraints, we consider a Ψ value of 0.3 corresponding to the last three columns in Table 11. On the other hand, we observe that the circuitry constraints are largely much tighter in the final solutions.

In general, the circuitry constraints are more difficult to satisfy when the value of Ω decreases. We observe in Table 11 that the solution times with Ψ–Ω values of 1.0–2.0 are greater than the ones with 1.0–3.0. As mentioned in Section IV.1.4.1, circuitry constraints with a lower Ω value, especially, may easily lead to infeasibility in solving subproblem on the RP-network provided by the master problem. That is, the RP-network may not contain any shortest paths satisfying the percentage circuitry constraints for some commodities. We note that, in these cases with infeasibility, it is also more difficult to find feasible solutions in the heuristic algorithm, and, in turn, to obtain good upper bounds to strengthen the cuts (4.43) in the ε-optimal approach. Furthermore, comparing the average runtimes with Ψ–Ω values of 1.0–3.0 versus 0.3–3.0, we observe that lower allowable load-imbalance levels generally increase total runtimes. However, as discussed above, the increase in the total runtime in this case is largely attributed to the increases in MP runtime, since the imbalance constraints are included in the MP.

In the sixth experiment, we examine the impact from different commodity distributions and the configuration of demand points. Two additional problem classes, Ui1 and Ui2, each with 10 instances, are generated. Classes Ui1 and Ui2 have $|\mathcal{N}|$ value of 80 with D values of 20 and 40, respectively. Their commodities consist of 40% long distance, 30% medium distance, and 30% short distance demand. Recall that classes Uf1 and Uf2 have the same characteristics as Ui1 and Ui2, respectively, except that the demands are for 60% long distance, 20% medium distance, and 20% short distance for Uf1 and Uf2. To examine a clustered configuration of demand origin/destination points in the region, we also generate problem classes Cf1, Cf2, Ci1, and Ci2 (each with 10 instances). Recall that the classes Cf1/Ci1 and Cf2/Ci2 have the same characteristics (number of nodes and commodities) as classes Uf1/Ui1 and Uf2/Ui2, except the, in Class "C", the nodes are clustered. The generation of

clustered instances is discussed in Section I.3.1.

We solve these instances without the circuitry restriction and display the results in Table 12. For differences in commodity distance distribution, we compare classes Uf1-2 to Ui1-2 and observe that Uf1 and Uf2 require slightly longer solution times than classes Ui1 and Ui2; in most cases, however, the differences are not significant. On the other hand, the distribution of origin/destination points can potentially affect the algorithmic performance. In particular, clustering of nodes restricts the connectivity of the underlying network, which in turn, reduces the number of feasible solutions. Thus, good feasible solutions are more difficult to find (indicated by the increased number of iterations - columns 4 vs 9) and constraint (4.43) can be weakened. As a result, the clustered instances take longer to solve (column 3 vs column 8). In terms of the percentage circuitry, fewer RPs are located in the clustered instances and, consequently, higher circuitry levels are reported. An increasing number of shorter distance commodities (Ui1 vs Uf1 and Ui2 vs Uf2) can also increase the circuitry level. The short (and sometimes medium) distance commodities are usually in the same region. These commodities can experience high circuitry levels if both the origin and destination are not RPs; they appear as good candidates for direct shipments to be determined at an operational level.

One instance – in class Cf1 with Δ_1-Δ_2-Ψ values of 20-40-0.2 – requires a significantly longer solution time than the other instances. This is due to the difficulty of finding a good feasible solution, and the master problem taking a very long time to verify infeasibility in the last iteration. Thus, the results reported for this particular class and setting are the average of 9 instances (solution time is listed in italics).

Table 12: Results for non-clustered and clustered instances

Δ_1-Δ_2-Ψ	Class	Non-clustered instances				Class	Clustered instances					
		Ave Time	Ave *Iter*	$\bar{\Omega}^M$	$\bar{\Omega}^A$		Ave Time	Ave *Iter*	$\bar{\Omega}^M$	$\bar{\Omega}^A$	L^A	$\bar{d}/L^A$
20-40-1	Uf1	867	4.1	10.1	0.19	Cf1	1663	6.3	15.8	0.24	3.8	22
20-50-1		477	2.7	9.4	0.18		712	3.6	14.4	0.22	3.4	24
30-50-1		856	3.9	12.5	0.22		1259	5.2	19.7	0.25	3.3	24
30-60-1		561	2.9	12.4	0.23		727	3.6	17.7	0.25	3.2	26
20-40-0.2		927	4.5	12.5	0.21		*1567*[1]	6.2	13.9	0.25	3.8	22
20-50-0.2		492	2.8	11.8	0.20		789	4.0	13.1	0.22	3.3	24
30-50-0.2		949	4.2	13.0	0.23		1613	6.2	22.0	0.26	3.3	25
30-60-0.2		660	3.5	10.4	0.24		830	4.0	19.8	0.25	3.2	26
20-40-1	Uf2	1266	3.1	9.8	0.13	Cf2	1643	3.8	16.2	0.19	3.8	21
20-50-1		809	2.1	11.3	0.12		1155	2.9	14.5	0.17	3.4	24
30-50-1		1374	3.1	15.0	0.16		1642	3.6	17.1	0.18	3.3	24
30-60-1		1134	2.7	15.4	0.16		1197	2.9	16.7	0.18	3.1	25
20-40-0.2		1293	3.2	12.5	0.14		1862	4.3	19.1	0.20	3.8	21
20-50-0.2		950	2.5	12.3	0.14		1315	3.3	11.6	0.18	3.4	24
30-50-0.2		1662	3.2	14.5	0.17		2071	4.4	18.3	0.19	3.3	23
30-60-0.2		1588	3.1	16.3	0.17		1396	3.4	20.0	0.19	3.1	25
20-40-1	Ui1	869	4.3	19.1	0.24	Ci1	1127	5.2	17.6	0.30	4.1	21
20-50-1		466	2.7	16.2	0.24		661	3.6	17.1	0.29	3.7	23
30-50-1		829	4.0	21.5	0.30		912	4.4	26.9	0.34	3.6	23
30-60-1		522	2.8	24.0	0.30		640	3.5	27.0	0.35	3.3	24
20-40-0.2		1457	5.2	17.2	0.27		1867	6.8	22.5	0.34	4.1	21
20-50-0.2		537	3.1	16.7	0.27		895	4.6	19.1	0.31	3.7	23
30-50-0.2		1070	4.4	22.4	0.30		1030	4.6	28.4	0.37	3.6	24
30-60-0.2		597	3.2	19.3	0.30		731	3.6	30.8	0.36	3.3	24
20-40-1	Ui2	1129	2.9	14.6	0.16	Ci2	1321	3.3	20.0	0.24	4.2	20
20-50-1		718	2.0	14.7	0.16		1106	3.0	22.4	0.22	3.6	22
30-50-1		1258	3.0	18.4	0.19		1440	3.5	20.4	0.24	3.6	22
30-60-1		813	2.1	18.0	0.18		1180	3.0	21.2	0.23	3.4	23
20-40-0.2		1211	3.1	14.8	0.17		1526	3.8	20.3	0.25	4.2	20
20-50-0.2		755	2.1	13.2	0.18		1167	3.2	22.2	0.23	3.6	22
30-50-0.2		1315	3.1	17.6	0.20		1381	3.4	24.3	0.26	3.7	22
30-60-0.2		970	2.5	17.9	0.19		1193	3.0	23.5	0.25	3.3	23

[1] The average of 9 instances without the outlier.

To present a comparison between the driving distances for the RP-network and PtP dispatching cases, we also record the average number of legs (L^A) and the average distance per leg ($\bar{d}/L^A$) from the results of the clustered instances (since they pose higher circuitry levels) in the last two columns of Table 12. In general, the

commodities are relayed by 3-4 legs on average and the average distance per leg is between 20-26. The results also show that, as Δ_1 and Δ_2 increase, the number of legs decreases, while the distance per leg increases. For these instances, the average direct shipment distance (calculated conservatively as Euclidean distance) for the commodities are 79.13, 79.23, 66.77 and 66.76 for classes Cf1-2 and Ci1-2, respectively. Since a driver, while handling a leg in an RP-network case, handles a commodity (from origin to destination) in the PtP-dispatching case, using an RP-network can shorten driver tour lengths by more than 50%, as observed when the direct shipment distances and $\bar{d}^A$ values are compared. This comes at the expense of an average 19-37% circuitry. The reduction in tour lengths can be even more pronounced when the assignment of drivers to consecutive direct shipments in PtP-dispatching is considered.

We also note that, although large $\bar{\Omega}^M$ values are reported, the majority of commodities have relatively low circuitry levels as indicated by very low $\bar{\Omega}^A$ values. Large additional distance and a high circuitry level normally occur when both the origin and destination of a very short distance commodity are not RPs so that the commodity would visit at least one RP on its trip. These high circuitry commodities are good candidates for the direct shipments; thus, they can be taken out and handled separately after a solution is obtained. Moreover, to facilitate better control of this large circuitry issue, we can include the circuitry constraints and resolve the problem as in the previous experiment.

In the seventh experiment, we compare the RP-networks of Model 1 to the networks of the uncapacitated single assignment hub location problem (SAHLP). To do this, we generate 10 instances of class Ue1, solve Model 1 to 2% optimality, and report the results of both models in Table 13; the values of Ψ and Ω are set to 0.3 and 3.0, and Δ_1–Δ_2 are fixed at 20–40.

Due to the lack of distance constraints and the complete graph assumption

in SAHLP, truck drivers are allowed to travel directly between commodities' origins/destinations and hubs. Thus, only a few hubs are required, and SAHLP network construction cost (Cost^{RP}) is smaller than in Model 1; the numbers of RPs are represented by #RP. Coupled with the total transportation cost (Cost^{T}) being approximately the same in both models, the Total Cost (Cost^{RP}+Cost^{T}) of SAHLP is lower than that of our Model 1. In addition, more flows enter and leave hubs, which consequently lowers the maximum and average load-imbalance levels ($\bar{\Psi}^{M}$ and $\bar{\Psi}^{A}$).

Table 13: Comparison between SAHLP and Model 1

Model	Total Cost		Cost^{RP}	Cost^{T}	#RP	d^{A}
SAHLP	1062724		68889	993835	7	81
Model 1	1181180		190000	991180	19	
Model	$\bar{\Psi}^{M}$	$\bar{\Psi}^{A}$	$\bar{\Omega}^{M}$	$\bar{\Omega}^{A}$	$\bar{d}/\text{L}^{M}$	$\bar{d}/\text{L}^{A}$
SAHLP	0.18	0.09	12.17	0.31	75.4	34.3
Model 1	0.26	0.14	2.78	0.20	38.8	23.0

However, Model 1 is more favorable than the SAHLP in terms of the maximum and average mileage per leg ($\bar{d}/\text{L}^{M}$ and $\bar{d}/\text{L}^{A}$), and the maximum and average percentage circuitry ($\bar{\Omega}^{M}$ and $\bar{\Omega}^{A}$). With fewer hubs in the intermediate locations and the direct transportation between origins/destinations and hubs, drivers must travel long distances before returning to home base and SAHLP provides inferior distance per leg. By comparing d^{A} to $\bar{d}/\text{L}^{A}$, we observe that SAHLP reduces the average tour length per leg (d^{A}) by 58% (over the direct shipment approach), while Model 1 reduces the tour length per leg by as much as 72%. Moreover, if neither of the commodities' origins and destinations are hubs, then drivers must travel on very circuitous paths as indicated by large circuitry levels.

Given the longer additional distances but smaller reduction in tour length per

leg, the SAHLP is less effective than Model 1, in terms of controlling drivers' tour lengths. In addition, Model 1 provides significant improvements in controlling both the average percentage circuitry and tour length per leg while only slightly increases the total cost (about 11% in this experiment).

In the eighth experiment, we compare the performance of Model 1's RP-networks and the networks obtained from the "Zone model," as presented in Taylor et al. (2001), using one instance of class Ue1. In this case, Model 1 is solved with different combinations of Δ_1, Δ_2, Ψ, and Ω. For the Zone model, we first note that Taylor et al. (2001) divide the entire service region into 5 zones based on the sales regions of J.B. Hunt Transport Inc. Due to the lack of such information, we partition the entire 150x100 service region into $K = 2$, 3, 4, 5, or 6 zones, as illustrated in Figure 4. We also note that two 4-zone models ($K = 4$) are considered in this experiment; one of them has 4 identical zones, and the other has zones of different sizes and shapes.

In Figure 4, each dot represents a node (a commodity's origin or destination, or the potential RP location), and is equipped with the location number. The zones' border lines are represented by black solid lines and all nodes inside the 20 miles bands (represented by dashed lines) along the zones' border lines have an RP located on them. Non-RP nodes in the interior of each zone have fixed membership in the zone in which they are located. On the other hand, the assignment of each RP to zones that share borders must be determined in order to obtain the actual zone's border (defined by the furthest RP locations). For illustration purposes, the RP locations and the possible assignments of each Zone model are summarized in Table 14.

Figure 4: Different Zone Models

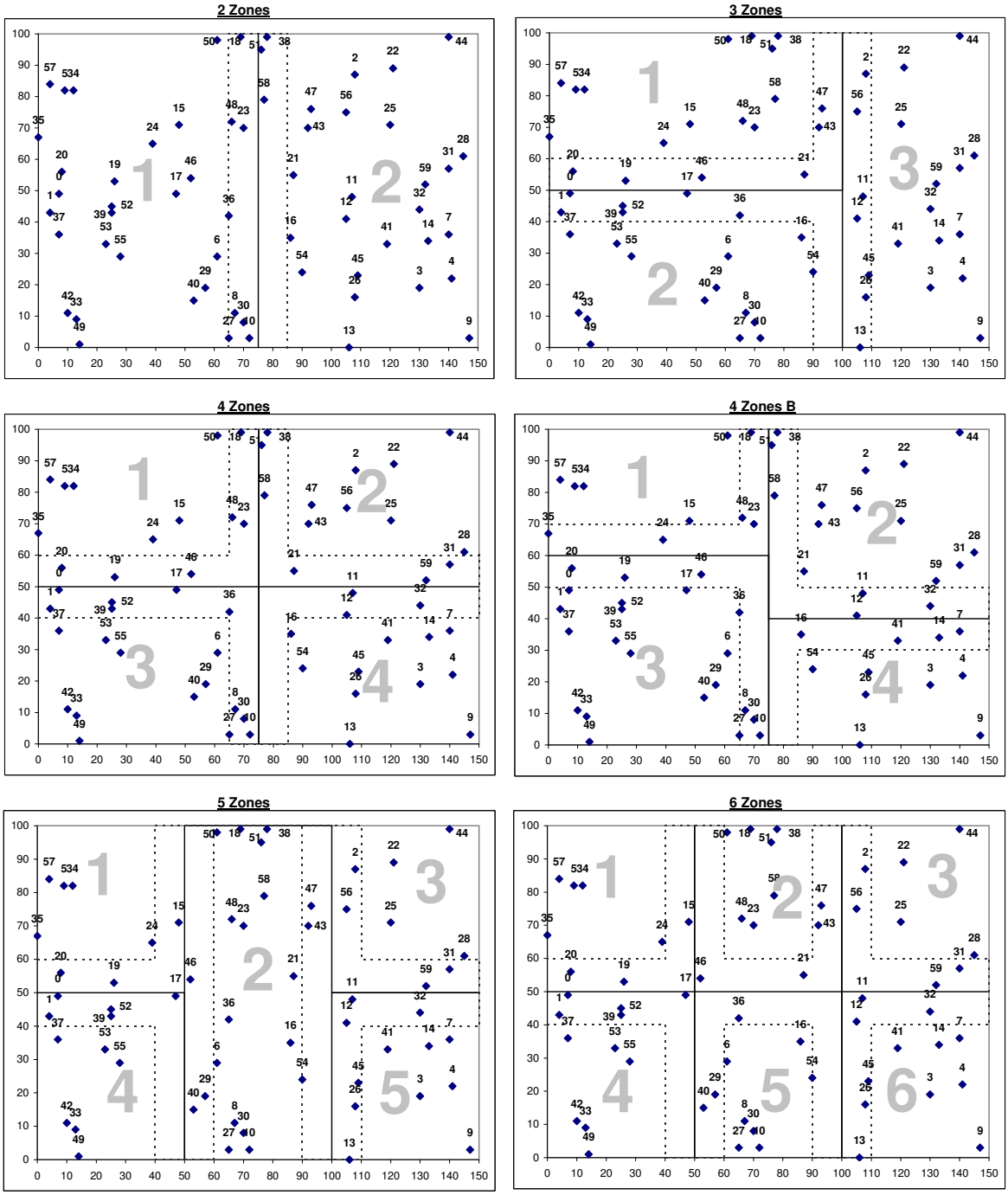


Table 14: Candidate zones of each RP in Zone models

Zone model	Candidate zones	RPs
2	1,2	8 , 10 , 18 , 23 , 27 , 30 , 36 , 38 , 48 , 51 , 58
3	1,2	0 , 1 , 17 , 19 , 20 , 21 , 36 , 39 , 46 , 52
	1,3	2 , 43 , 47 , 56
	2,3	13 , 26 , 45 , 54
	1,2,3	11 , 12
4	1,2	18 , 23 , 38 , 48 , 51 , 58
	1,3	0 , 1 , 17 , 19 , 20 , 36 , 39 , 46 , 52
	2,4	11 , 12 , 21 , 31 , 32 , 59
	3,4	8 , 10 , 27 , 30
4 B	1,2	18 , 38 , 48 , 51 , 58
	1,3	19 , 20 , 24 , 35 , 46
	2,4	7 , 11 , 12 , 14 , 16 , 32 , 41
	3,4	8 , 10 , 27 , 30
	1,2,3	23
	2,3,4	36
5	1,2	15
	1,4	0 , 1 , 19 , 20 , 39 , 52
	2,3	2 , 43 , 47 , 56
	2,4	29 , 40 ,
	2,5	13 , 26 , 45 , 54
	3,5	31 , 32 , 59
	1,2,4	17 , 46
	2,3,5	11 , 12
6	1,2	15
	1,4	0 , 1 , 19 , 20 , 39 , 52
	2,3	2 , 43 , 47 , 56
	2,4	21 , 36
	3,6	31 , 32 , 59
	4,5	29 , 40
	5,6	13 , 26 , 45 , 54
	1,2,4,5	17 , 46
	2,3,5,6	11 , 12

The Zone model constructs RP-networks in such a way that the load imbalances between zones are small. To do this, the assignment of each RP is determined by solving the following formulation, where ZP_k and ZN_k are the decision variables corresponding to the positive and negative load-imbalance levels of zone k:

$$\text{Min} \quad \sum_k ZP_k - \sum_k ZN_k \tag{4.51}$$

subject to

$$\sum_i \Big(\sum_j (w_{ji} - w_{ij}) \Big) x_{ik} + ZP_k - ZN_k = 0 \quad \forall\, k \in \{1, \ldots, K\} \tag{4.52}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{4.53}$$

$$x_{ik} \in \{0, 1\},\ ZP_k \geq 0,\ ZN_k \leq 0 \qquad \forall\, i \in \mathcal{N}, k \in \{1, \ldots, K\} \tag{4.54}$$

The objective function (4.51) minimizes the total load-imbalance level over zones. Constraints (4.52) determine the level of the load-imbalance for the zone. Constraints (4.53) ensure that every RP is assigned to only one zone. Constraints (4.54) state the binary requirement of $\mathbf{x}$ variables, and that ZP_k (ZN_k) is a non-negative (non-positive) real number. After solving the formulation (4.51)-(4.54) to obtain the assignment of RPs to zones, every commodity is routed from the origin – through one of the RPs in the zone to which the origin is assigned – directly to the destination. We note that the zone model does not consider the single assignment of nodes to RPs; therefore, loads originating from the same origin can be routed through different RPs. The results and solution statistics of the Zone model and Model 1 are reported in Table 15.

In Table 15, the results show that, for this particular instance, every setting of Model 1 performs better than the Zone models in terms of total cost. This is largely

Table 15: Comparison between Model 1 and Zone models

	Δ_1-Δ_2-Ψ-Ω	Total Cost	Cost^{RP}	Cost^{T}	#RP	d^A
Model 1	30-50-0.3-x	1155160	140000	1015160	14	83.60
	30-50-0.3-3	1159120	150000	1009120	15	
	30-50-1-x	1149430	140000	1009430	14	
	30-50-1-0.3	1152400	150000	1002400	15	
	30-60-0.3-x	1136800	120000	1016800	12	
	30-60-0.3-3	1145480	140000	1005480	14	
	30-60-1-x	1132720	120000	1012720	12	
	30-60-1-3	1144400	140000	1004400	14	
	Zone	Total Cost	Cost^{RP}	Cost^{T}	#RP	d^A
Zone Models	2	1165010	110000	1055010	11	83.60
	3	1163770	200000	963770	20	
	4	1181730	250000	931727	25	
	4B	1244630	230000	1014630	23	
	5	1198650	240000	958651	24	
	6	1209170	260000	949166	26	

	Δ_1-Δ_2-Ψ-Ω	$\bar{\Psi}^M$	$\bar{\Psi}^A$	$\bar{\Omega}^M$	$\bar{\Omega}^A$	$\bar{d}/\text{L}^M$	$\bar{d}/\text{L}^A$
Model 1	30-50-0.3-x	0.30	0.13	9.6	0.24	49.0	26.9
	30-50-0.3-3	0.29	0.13	3.0	0.20	49.0	26.3
	30-50-1-x	0.34	0.14	10.4	0.23	49.0	27.0
	30-50-1-0.3	0.47	0.15	3.0	0.18	48.0	25.7
	30-60-0.3-x	0.27	0.12	8.3	0.24	59.0	29.3
	30-60-0.3-3	0.25	0.12	3.0	0.19	55.5	27.5
	30-60-1-x	0.34	0.13	10.4	0.24	59.0	29.0
	30-60-1-3	0.34	0.13	3.0	0.19	59.0	27.5
	Zone	$\bar{\Psi}^M$	$\bar{\Psi}^A$	$\bar{\Omega}^M$	$\bar{\Omega}^A$	$\bar{d}/\text{L}^M$	$\bar{d}/\text{L}^A$
Zone Models	2	0.97	0.56	39.0	0.52	84.5	48.8
	3	0.96	0.65	12.1	0.19	82.5	44.7
	4	1.00	0.72	3.3	0.08	83.0	43.3
	4B	1.00	0.75	56.0	0.31	93.5	47.0
	5	1.00	0.67	10.6	0.16	92.5	44.5
	6	1.00	0.79	7.3	0.10	82.5	44.0

due to a larger number of RPs being located in the Zone model, which leads to a very high network construction cost (Cost^{RP}). However, in the 2-zone model, where only 11 RPs are located, the transportation cost (Cost^{T}) becomes quite large and the total cost remains high. Not only can too many RPs be located, a network with poor configuration (RP locations) can be obtained by manually selecting RP locations; this is illustrated by the 4B-zone model. Although a large number of RPs (23) are located in the 4B-zone model, and once the commodities arrive at the RPs, they can travel directly to their destinations, the transportation cost is still higher than most settings of Model 1. Thus, it is clear that, for the Zone model to perform well in terms of total cost, one must carefully balance network construction cost and transportation cost, a process that is automatically handled in Model 1.

For operational efficiency, we observe that Model 1 has smaller maximum and average load-imbalance levels ($\bar{\Psi}^{M}$ and $\bar{\Psi}^{A}$) than in the Zone model. Although the zones' load-imbalance levels are minimized in the zone model, the RPs' load-imbalance levels are not minimized, creating a greater difference between the entering and leaving loads at every RP. This is mainly due to RPs only being used by outgoing commodities in the inter-zone transportation; incoming loads do not visit the zone's RPs before arriving at destinations. Such an imbalance would lead to local drivers having a high level of empty mileage. We also observe that Model 1 provides a better control of the percentage circuitry level, especially when the percentage circuitry constraints are used (if the percentage circuitry constraints are not used, then it is indicated by "x"); the maximum and average percentage circuity levels ($\bar{\Omega}^{M}$ and $\bar{\Omega}^{A}$) of the Zone model fluctuate highly. In terms of tour length reduction, Model 1 reduces average tour length (distance per leg) by approximately 65-69% (compare d^{A} with $\bar{d}/\text{L}^{A}$), whereas the Zone model reduces it by only 42-48%. Moreover, the maximum and average local drivers' tour lengths, $\bar{d}/\text{L}^{M}$ and $\bar{d}/\text{L}^{A}$ of the Zone model are almost twice those

from Model 1. In fact, when Δ_2 is set to 50, the maximum tour lengths of Model 1 are only slightly larger than the average tour lengths of the Zone model.

Based on these results, we observe that Model 1 provides better control of the load-imbalance (empty mileage), percentage circuitry (additional distance), and distance per leg (tour length) over the Zone model. At the same time, it is capable of generating a more cost efficient RP-network.

IV.1.6. Concluding Remarks

To minimize the sum of transportation and fixed relay points location costs, while satisfying tour length, load-imbalance, and percentage circuitry constraints, Model 1 determines 1) the relay point locations, 2) the assignment of nodes to the relay points, and 3) the routes of the TLs from their origins to their destinations. We observe, in an MIP formulation of Model 1, that for given relay locations and node assignments to relays, the remaining problem can be posed as a linear program. This facilitates a solution approach based on Benders decomposition. Due to inefficiencies in solving our problem via a typical implementation of Benders decomposition, we explore several avenues for algorithmic improvement in a systematic fashion.

Specifically, we develop an approach for deriving strengthened Benders cuts to accelerate the algorithm convergence and reduce the total runtime. In addition, we enhance the performance of the algorithm through the use of cut disaggregation schemes and surrogate constraints which promote a tighter lower bound and can help reduce the number of iterations. Observing the rapid growth in master problem runtimes with the use of disaggregated cuts, we also employ the ε-optimal framework which helps to improve the computational effort in solving the master problem. Furthermore, for the purpose of strengthening the upper bound, we design a local search heuristic with effective neighborhood functions that provide improved feasible solu-

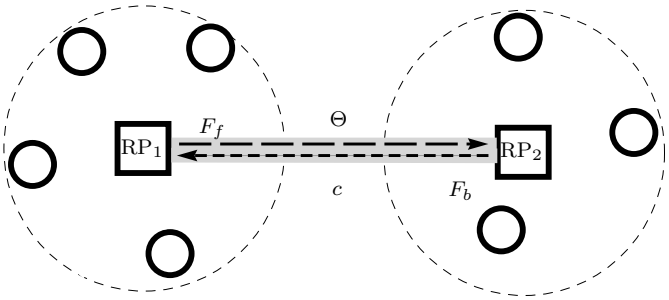
tions (upper bounds). Our computational results illustrate significant improvement using our solution algorithms as opposed to the typical Benders decomposition and the Branch-and-cut approach. In testing our algorithms, we conduct experiments to observe the effect of input parameters on the solution algorithms. Furthermore, we also compare RP-networks obtained from Model 1 to the networks from other related models in the literature to illustrate Model 1's efficacy in controlling the tour lengths and other performance metrics.

IV.2. Model 2: RNDP with Link-Imbalance and Link-Capacity Constraints

In this section, we extend the base model to include both the link-imbalance and the link capacity considerations. Contrary to the load imbalance constraints that aim to control local drivers' empty mileage, "link-imbalance constraints" focus more on lane truck drivers. Recall that the load-imbalance requirement in Model 1 provides the RP-network with a balanced number (small difference) of local pick-ups and deliveries to facilitate control of local drivers' empty mileage. However, there is no guarantee that the local empty mileage in such a network will be small. On the other hand, link-balance constraints control lane drivers' empty mileage more directly by keeping the difference between the forward and backward flow to a low level. For illustration purposes, we use Figure 5 to show the connection between the link-imbalance and lane drivers' empty mileage discussed herein.

Between the RP_1 and RP_2 in Figure 5, F_f represents forward flow from RP_1 to RP_2 whereas F_b represents backward flow. The link-imbalance constraints require the percentage difference between F_f and F_b to be within the permissible link-imbalance level Θ. That is, if $F_f > F_b$, then $F_f - F_b$ must not be greater than $\Theta \times F_f$. If

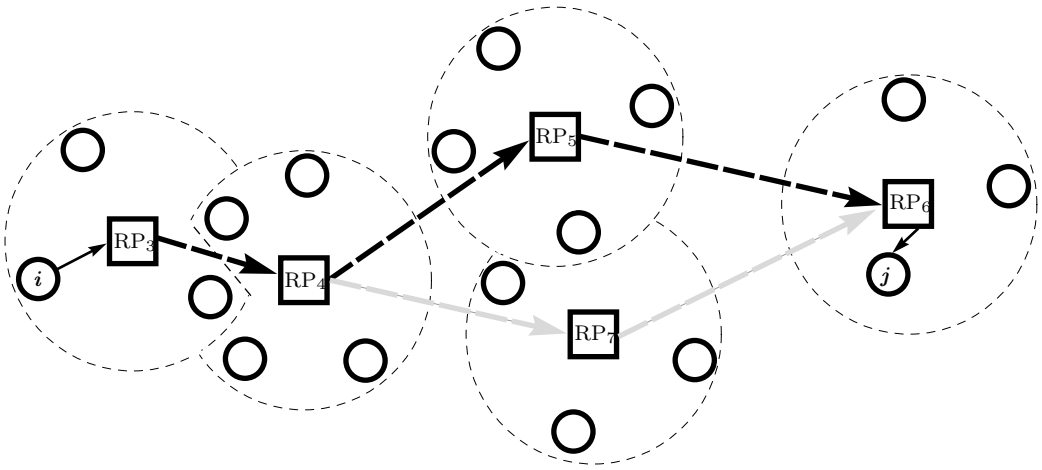
Figure 5: Link-Imbalance and Capacity Constraints



we assume that truck drivers must travel back to the RP from which the drivers are dispatched, then $F_f - F_b$ is the number of empty back hauls and the total number of trips is $2 \times F_f$. In this case, controlling the link-imbalance to be within Θ, $\frac{F_f - F_b}{F_f} \leq \Theta$ directly implies that empty mileage does not exceed $\frac{\Theta}{2}$ as $\frac{F_f - F_b}{2 \times F_f} \leq \frac{\Theta}{2}$. According to Erlbaum and Holguín-Veras (2006), the empty truck miles of the trip are typically between 24-33% with an average value of 27.47%. Therefore, setting Θ value to 60% (for round trip) or 0.6 ensures that, in the worst case, the empty mileage from the RP-network is comparable to current industry averages (Θ value of 0.6 is equivalent to 30% empty mileage which is only slightly above 27.47%). In addition to the link-imbalance constraints, Model 2 also considers the "link capacity constraints" that limit the total flow between every pair of RPs. In Figure 5, the total of F_f and F_b cannot exceed link capacity c.

In summary, Model 2 determines 1) RP locations, 2) nonRP nodes assignment, and 3) the actual route for each commodity to minimize the total RP location cost and total commodity transportation cost, and satisfies tour length, link-imbalance, and link capacity constraints. The general characteristics of Model 2 are the same as in the base model and Model 1. However, due to limited link capacity, a commodity

Figure 6: A Schematic View of Model 2



may not travel the shortest possible path between the RP of the origin and the RP of the destination. As illustrated in Figure 6, fractions of the commodity $[i, j]$ may be routed through RP_3-RP_4-RP_5-RP_6 and RP_3-RP_4-RP_5-RP_7-RP_6 if the capacity between the RP_4-RP_5 link is exhausted.

IV.2.1. The Model

Following the above discussion, we utilize the notation presented in Chapter III to define the link-imbalance as:

$$\Big| \sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \Big| \leq \Theta \max \Big\{ \sum_i \sum_j w_{ij}\, y_{kl}^{ij}, \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \Big\} \quad \forall\, k, l \in \mathcal{N} \tag{4.55}$$

In constraints (4.55), the terms $\sum_i \sum_j w_{ij}\, y_{kl}^{ij}$ and $\sum_i \sum_j w_{ij}\, y_{lk}^{ij}$ correspond to the total forward and backward flows on a pair of RPs (k, l). The flow difference must be within Θ percent of the larger of the two flows. Note that constraints (4.55) are

non-linear and can be stated in a linear form as:

$$\sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \qquad \forall\, k, l \in \mathcal{N} \tag{4.56}$$

$$\sum_i \sum_j w_{ij}\, y_{lk}^{ij} - \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \qquad \forall\, k, l \in \mathcal{N} \tag{4.57}$$

However, it is clear that some constraints in (4.56) and (4.57) are the same (e.g., constraint (4.56) for link $(1, 2)$ is the same as constraint (4.57) for link $(2, 1)$). Thus, stating the link-imbalance constraints as in (4.58) and (4.59) can reduce the number of constraints in the formulation.

$$\sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \qquad \forall\, k, l \in \mathcal{N},\, k < l \tag{4.58}$$

$$\sum_i \sum_j w_{ij}\, y_{lk}^{ij} - \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \qquad \forall\, k, l \in \mathcal{N},\, k < l \tag{4.59}$$

In addition, the capacity constraints of Model 4 are stated as follows:

$$\sum_i \sum_j w_{ij}\, (y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl} \qquad \forall\, k, l \in \mathcal{N},\, k < l \tag{4.60}$$

In constraints (4.60), the total flow in both directions must not exceed the link capacity c_{kl}.

By incorporating constraints (4.58), (4.59), and (4.60) into the base model, the complete formulation of Model 2 is as follows:

$$\text{Min } Z = \sum_i \sum_k T_1 \, d_{ik} \sum_j (w_{ij} + w_{ji}) \, x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2 \, d_{kl} \, w_{ij} \, y_{kl}^{ij} + \sum_k F_k \, x_{kk} \tag{4.61}$$

subject to

$$d_{ik} \, x_{ik} \leq \Delta_1 \qquad \forall \, i, k \in \mathcal{N} \tag{4.62}$$

$$d_{kl} \, z_{kl} \leq \Delta_2 \qquad \forall \, k, \, l \in \mathcal{N}, \, k < l \tag{4.63}$$

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = x_{jk} - x_{ik} \qquad \forall \, [i, j] \in \mathcal{Q}, \, \forall \, k \in \mathcal{N} \tag{4.64}$$

$$\sum_i \sum_j w_{ij} \, y_{kl}^{ij} - \sum_i \sum_j w_{ij} \, y_{lk}^{ij} \leq \Theta \sum_i \sum_j w_{ij} \, y_{kl}^{ij} \qquad \forall \, k, l \in \mathcal{N}, \, k < l \tag{4.65}$$

$$\sum_i \sum_j w_{ij} \, y_{lk}^{ij} - \sum_i \sum_j w_{ij} \, y_{kl}^{ij} \leq \Theta \sum_i \sum_j w_{ij} \, y_{lk}^{ij} \qquad \forall \, k, l \in \mathcal{N}, \, k < l \tag{4.66}$$

$$\sum_i \sum_j w_{ij} \, (y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl} \qquad \forall \, k, l \in \mathcal{N}, \, k < l \tag{4.67}$$

$$\sum_k x_{ik} = 1 \qquad \forall \, i \in \mathcal{N} \tag{4.68}$$

$$x_{ik} \leq x_{kk} \qquad \forall \, i, \, k \in \mathcal{N} \tag{4.69}$$

$$z_{kl} \leq x_{kk} \qquad \forall \, k, \, l \in \mathcal{N}, \, k < l \tag{4.70}$$

$$z_{kl} \leq x_{ll} \qquad \forall \, k, \, l \in \mathcal{N}, \, k < l \tag{4.71}$$

$$y_{kl}^{ij} \leq z_{kl} \qquad \forall \, [i, j] \in \mathcal{Q}, \, \forall \, k, l \in \mathcal{N}, \, k < l \tag{4.72}$$

$$y_{lk}^{ij} \leq z_{kl} \qquad \forall \, [i, j] \in \mathcal{Q}, \, \forall \, k, l \in \mathcal{N}, \, k < l \tag{4.73}$$

$$x_{ik}, z_{kl} \in \{0, 1\}, \, 0 \leq y_{kl}^{ij} \leq 1 \qquad \forall \, i, \, j, \, k \in \mathcal{N} \tag{4.74}$$

Without the link-imbalance constraints (4.65) and (4.66) and the link capacity constraints (4.67), Model 2 is actually the same as the base model. The objective function (4.61), the tour length constraints (4.62), the flow conservation constraints (4.64), the single assignment constraints (4.68), and the construction constraints (4.69) are identical to those of the base model and Model 1. The only differences are the use of binary **z** variables in the lane tour length constraints (4.63) and in the constraints defining the structural requirements of RP-networks (4.13)-(4.15). Specifically, constraints (4.70)-(4.73) serve the same objective as constraints (2.7)-(2.8) in the base model and constraints (4.13)-(4.15) in Model 1.

IV.2.2. Lagrangean Decomposition Framework

Although Benders decomposition is applicable to Model 2, its performance is not as promising as when it was applied to Model 1 in Section IV.1. Even with given **x** variables, the link-imbalance and the link capacity constraints prevent us from decomposing the **y** subproblem for each commodity. In addition, since **y** variables account for the majority portion of Model 2, solving the **y** subproblem in its non-decomposed form is inefficient. All these disadvantages restrict us from solving Model 2 with Benders decomposition.

After a close examination of the underlying structure of Model 2, we observe that if the connection between **x** and **y** variables is removed or relaxed, then the subproblem containing only **y** variables can regain its decomposable property. Such relaxation in Model 2 can be achieved using a Lagrangean decomposition (LD) framework. Model 2 is also amenable to solution by the Lagrangean relaxation (LR) framework, as presented in Section V.2.2. However, due to the large number of constraints that must be relaxed in order to decompose the relaxed problem (which will lead to weak lower bounds), we develop the solution algorithms for Model 2 based on the LD

framework. Thus, in this section, we provide a detailed discussion of the development of our LD based algorithms.

As invented in Guignard and Kim (1987), Lagrangean decomposition is a relaxation technique that utilizes Lagrangean multipliers to decompose a large mathematical model into two relatively smaller and less computationally intensive subproblems. In LD, a number of original decision variables are replaced by a set of duplicated variables linked together by the "copy constraints". This forces the original and duplicated variables to have equal values. The relaxation of the copy constraints enables the decomposition of the model. Guignard and Kim (1987) applied the LD framework to scheduling problems and reported its ability to provide very tight lower bounds.

Prior to applying the LD framework, we first observe that constraints (4.62) can be removed after assigning zero values to the x_{ik} variables whose distance on the arc (i, k) is greater than Δ_1. Likewise, constraints (4.63) can also be removed after assigning zero values to the z_{kl} and y_{kl}^{ij} variables whose distance on the arc (k, l) is greater than Δ_2. We refer to this preprocessed formulation without constraints (4.62)-(4.63) as "RPND$_{xyz}$". We also note that the $\mathbf{z}$ variables, which prevent lane drivers from traveling further than Δ_2, can be removed from the formulation if constraints (4.70)-(4.73) are replaced by constraints (4.75)-(4.76).

$$y_{kl}^{ij} \leq x_{kk} \qquad \forall\, [i, j] \in \mathcal{Q},\, \forall\, k,\, l \in \mathcal{N} \tag{4.75}$$

$$y_{kl}^{ij} \leq x_{ll} \qquad \forall\, [i, j] \in \mathcal{Q},\, \forall\, k,\, l \in \mathcal{N} \tag{4.76}$$

We now refer to the RPND$_{xyz}$ model with constraints (4.75)-(4.76) and without constraints (4.70)-(4.73) as "RPND$_{xy}$".

IV.2.2.1. Copy Constraints and Modified Model

Based on the RPND_{xy} formulation given above, we define our copy constraints as follows:

$$u_k^{ij} = \sum_l y_{kl}^{ij} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.77}$$

$$v_k^{ij} = \sum_l y_{lk}^{ij} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.78}$$

$$0 \leq u_k^{ij}, v_k^{ij} \leq 1 \qquad \forall\, i,\, j,\, k \in \mathcal{N} \tag{4.79}$$

Due to the large size of the model and the interrelation between $\mathbf{x}$ and $\mathbf{y}$ variables, defining the copy constraints in this aggregated form not only facilitates the decomposition of RPND_{xy} but also helps control the formulation size. In order to decompose RPND_{xy}, we observe that the flow conservation constraints (4.64) can be restated using $\mathbf{u}$ and $\mathbf{v}$ variables as

$$v_k^{ij} - u_k^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.80}$$

Moreover, constraints (4.75) and (4.76) can also be aggregated as:

$$\sum_l y_{kl}^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.81}$$

$$\sum_l y_{lk}^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.82}$$

Since $u_k^{ij} = \sum_l y_{kl}^{ij}$ and $v_k^{ij} = \sum_l y_{lk}^{ij}$, then we can substitute the RHS of constraints (4.81) and (4.82) with u_k^{ij} and v_k^{ij} in order to separate $\mathbf{x}$ variables from $\mathbf{y}$

variables as follows:

$$u_k^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.83}$$

$$v_k^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.84}$$

Finally, our modified "RPND$_{xyuv}$" formulation can be stated as:

$$\text{Min } Z_{RPND} = \sum_i \sum_k T_1\, d_{ik} \sum_j (w_{ij}+w_{ji})\, x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} + \sum_k F_k\, x_{kk} \tag{4.85}$$

subject to

$$v_k^{ij} - u_k^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.86}$$

$$\sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \qquad \forall\, k \in \mathcal{N},\, k < l \tag{4.87}$$

$$\sum_i \sum_j w_{ij}\, y_{lk}^{ij} - \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \leq \Theta \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \qquad \forall\, k \in \mathcal{N},\, k < l \tag{4.88}$$

$$\sum_i \sum_j w_{ij}\, (y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{4.89}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{4.90}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,\, k \in \mathcal{N} \tag{4.91}$$

$$u_k^{ij} = \sum_l y_{kl}^{ij} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.92}$$

$$v_k^{ij} = \sum_l y_{kl}^{ij} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.93}$$

$$u_k^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.94}$$

$$v_k^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.95}$$

$$x_{ik} \in \{0,1\},\, 0 \leq y_{kl}^{ij}, u_k^{ij}, v_k^{ij} \leq 1 \qquad \forall\, i,\, j,\, k \in \mathcal{N} \tag{4.96}$$

IV.2.2.2. Decomposed Subproblems

There are several alternative relaxation approaches for the RPND$_{xyuv}$ model; however, based on the LD framework, we relax the copy constraints (4.92)-(4.93). To do so, we define τ_k^{ij} and σ_k^{ij} as the Lagrangean multipliers associated with constraints (4.92) and (4.93), respectively. Thus, by removing the copy constraints and incorporating them into the objective function (4.85), the objective function can be restated as:

$$\text{Min } Z_{RRND} = \sum_i \sum_k T_1\, d_{ik} \sum_j (w_{ij}+w_{ji})\, x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} + \sum_k F_k\, x_{kk}$$

$$+ \sum_i \sum_j \sum_k \tau_k^{ij} \left(\sum_l y_{kl}^{ij} - u_k^{ij}\right) + \sum_i \sum_j \sum_k \sigma_k^{ij} \left(\sum_l y_{lk}^{ij} - v_k^{ij}\right) \tag{4.97}$$

The relaxed model, with the objective function (4.97) and constraints (4.86)-(4.91) and (4.94)-(4.96), is now referred to as "RRND". Solving the RRND model with any value of $\boldsymbol{\tau}$ and $\boldsymbol{\sigma}$ provides a lower bound to the original problem where the last two terms of (4.95) can be viewed as the penalty arising from the violation of constraints (4.92) and (4.93). In this case, whenever the relaxed constraints are satisfied, then there is no penalty and the optimal solution to the RPND$_{xyuv}$ model is obtained.

Clearly, the RRND model can be decomposed into two subproblems, one with **x**, **u**, and **v** variables and one with only **y** variables. The subproblem "RRND$_{xuv}$" associated with the decision variables **x**, **u**, and **v** is as follows:

$$\text{Min } Z_{\mathbf{xuv}} = \sum_i \sum_k T_1\, d_{ik} \sum_j (w_{ij} + w_{ji})\, x_{ik} + \sum_k F_k\, x_{kk}$$

$$- \sum_i \sum_j \sum_k \tau_k^{ij}\, u_k^{ij} - \sum_i \sum_j \sum_k \sigma_k^{ij}\, v_k^{ij} \tag{4.98}$$

subject to

$$v_k^{ij} - u_k^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.99}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{4.100}$$

$$x_{ik} \le x_{kk} \qquad \forall\, i,\, k \in \mathcal{N} \tag{4.101}$$

$$u_k^{ij} \le x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.102}$$

$$v_k^{ij} \le x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{4.103}$$

$$x_{ik} \in \{0,1\},\, 0 \le v_k^{ij},\, u_h^{ij} \le 1 \qquad \forall\, i,\, j,\, k \in \mathcal{N} \tag{4.104}$$

The second subproblem, "RRND$_y$", associated with the $\mathbf{y}$ variables is as follows:

$$\text{Min} \qquad Z_{\mathbf{y}} = \sum_i \sum_j \sum_k \sum_l (T_2\, d_{kl}\, w_{ij} + \tau_k^{ij} + \sigma_l^{ij})\, y_{kl}^{ij} \tag{4.105}$$

subject to

$$\sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \le \Theta \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{4.106}$$

$$\sum_i \sum_j w_{ij}\, y_{lk}^{ij} - \sum_i \sum_j w_{ij}\, y_{kl}^{ij} \le \Theta \sum_i \sum_j w_{ij}\, y_{lk}^{ij} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{4.107}$$

$$\sum_i \sum_j w_{ij}\, (y_{kl}^{ij} + y_{lk}^{ij}) \le c_{kl} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{4.108}$$

$$0 \le y_{kl}^{ij} \le 1 \qquad \forall\, i,\, j,\, k \in \mathcal{N} \tag{4.109}$$

As aforementioned, the subproblem RRND$_y$ can be further decomposed for each RP-RP link (k,l), $k < l$, $k,l \in \mathcal{N}$. The decomposed subproblem is referred to as "RRND$_y^{kl}$" with the associated objective function value $Z_{\mathbf{y}}^{kl}$ where $\sum_k \sum_l Z_{\mathbf{y}}^{kl} = Z_{\mathbf{y}}$ and $Z_{\mathbf{y}} + Z_{\mathbf{xuv}} = Z_{RRND} \le Z_{RPND}$.

IV.2.2.3. Solving the Subproblems

The subproblem RRND_{xuv} is significantly smaller than the RPND_{xyuv} problem since $\mathbf{y}$ variables are not included, hence, solving RRND_{xuv} problem with the Branch-and-cut approach (CPLEX) now provides satisfactory performance. On the other hand, it is more computationally intensive to solve the RRND_y problem, which is essentially a continuous knapsack problem with side constraints (4.106)-(4.107). When constraints (4.106)-(4.107) are relaxed, the RRND_y^{kl} subproblem can be solved using the greedy algorithm presented in Algorithm 4.

Algorithm 4 Solving problem RRND_y^{kl} without the link-imbalance constraints

Set $Z_{\mathbf{y}}^{kl} = 0$, $C_{kl} = c_{kl}$, $y_{kl}^{ij} = y_{lk}^{ij} = 0, \forall\, [i,j] \in \mathcal{Q}$;
Let $B_{kl}^{ij} = \frac{T_2\, d_{kl}\, w_{ij} + \tau_k^{ij} + \sigma_l^{ij}}{w_{ij}}$ and $B_{lk}^{ij} = \frac{T_2\, d_{lk}\, w_{ij} + \tau_l^{ij} + \sigma_k^{ij}}{w_{ij}}$;
Sort B_{kl}^{ij} and B_{lk}^{ij} in ascending order;
while $C_{kl} > 0$ **do**
  $\hat{B}_{kl}^{ij} = \min\{B_{kl}^{ij}, B_{lk}^{ij}\}$;
  if $\hat{B}_{kl}^{ij} \geq 0$ **then**
    stop;
  else
    Let $(\hat{i}, \hat{j}, \hat{k}, \hat{l})$ be the indices associate with $\hat{B}_{kl}^{ij}$;
    $Z_{\mathbf{y}}^{kl} = Z_{\mathbf{y}}^{kl} + \min\{\hat{B}_{kl}^{ij}, (\hat{B}_{kl}^{ij} * \frac{C_{kl}}{w_{\hat{i}\hat{j}}})\}$;
    $C_{kl} = C_{kl} - \min\{C_{kl}, w_{\hat{i}\hat{j}}\}$;
    $y_{\hat{k}\hat{l}}^{\hat{i}\hat{j}} = y_{\hat{k}\hat{l}}^{\hat{i}\hat{j}} + \min\{1, \frac{C_{kl}}{w_{\hat{i}\hat{j}}}\}$
    Remove $\hat{B}_{kl}^{ij}$ from the list;
  end if
end while

Algorithm 4 tries to fill the available capacity of the link (k,l) with the commodity $[i,j]$ that has the largest negative objective function coefficient. For simplicity, we refer to the coefficient of the $\mathbf{y}$ variable as B_{kl}^{ij}. In this case, the commodity with the smallest B_{kl}^{ij} is augmented, one at a time, and the link capacity is adjusted accordingly.

If enough capacity C_{kl} is available, the associated $\mathbf{y}$ variable has the value of one; otherwise, only fractions of the demand are augmented and the associated $\mathbf{y}$ takes a value between 0 and 1. The augmentation process terminates when the link capacity is exhausted or when the smallest B_{kl}^{ij} is non-negative. We denote the solution obtained from this knapsack problem as $\hat{y}_{kl}$.

The link-imbalance constraints (4.106)-(4.107), which are relaxed in the greedy algorithm, may be violated If constraints (4.106)-(4.107) are satisfied, then $\hat{y}_{kl}$ is also the optimal solution for the RRND_y^{kl} subproblem. However, if that is not the case, then further modification is required to fix the obtained infeasible solution.

In order to fix an infeasible $\hat{y}_{kl}$, we first observe that only one direction of constraints (4.106)-(4.107) can be violated at a time; This is equivalent to having too many flows sent in the violated direction. Based on this observation, Algorithm 5 is developed for converting the infeasible $\hat{y}_{kl}$ into the optimal y_{kl}. Without loss of generality, we assume that constraint (4.106), corresponding to direction (k, l), is violated. Thus, $\sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij} > \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij}$ and $\sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij} - \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij} > \Theta(\sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij})$. We also note that, in Algorithm 5, if constraint (4.107) is violated, then every (k, l) must be changed to (l, k), and every (l, k) must be changed to (k, l).

Algorithm 5 Fixing infeasible $\hat{y}_{kl}$ for RRND_y^{kl}

1: $\hat{Z}_{\mathbf{y}}^{kl} = \sum_i \sum_j \sum_k \sum_l (T_2\, d_{kl}\, w_{ij} + \tau_k^{ij} + \sigma_l^{ij})\, \hat{y}_{kl}^{ij}$;
2: $\hat{C}_{kl} = c_{kl} - \sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij} + \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij}$;
3: Let $B_{kl}^{ij} = \frac{T_2\, d_{kl}\, w_{ij} + \tau_k^{ij} + \sigma_l^{ij}}{w_{ij}}$ and $B_{lk}^{ij} = \frac{T_2\, d_{lk}\, w_{ij} + \tau_l^{ij} + \sigma_k^{ij}}{w_{ij}}$;

4: **Remove TLs in the direction** (k,l) **until constraint (4.106) is satisfied with equality**
5: **while** $\sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij} - \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij} > \Theta(\sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij})$ **do**
6: Let $B_{kl}^{mn} = \max\{B_{kl}^{ij} : \hat{y}_{kl}^{ij} > 0\}$;
7: Let $W = (1-\Theta) \sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij} - \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij}$;
8: **if** $W > w_{mn} \hat{y}_{kl}^{mn}$ **then**
9: $\hat{Z}_{\mathbf{y}}^{kl} = \hat{Z}_{\mathbf{y}}^{kl} - B_{kl}^{mn} w_{mn} \hat{y}_{kl}^{mn}$; $\hat{C}_{kl} = \hat{C}_{kl} + w_{mn} \hat{y}_{kl}^{mn}$; $\hat{y}_{kl}^{mn} = 0$;
10: **else**
11: $\hat{Z}_{\mathbf{y}}^{kl} = \hat{Z}_{\mathbf{y}}^{kl} - B_{kl}^{mn} \big(\frac{w_{mn} - W}{w_{mn}}\big)$; $\hat{C}_{kl} = \hat{C}_{kl} + w_{mn} - W$; $\hat{y}_{kl}^{mn} = \frac{W}{w_{mn}}$;
12: **end if**
13: **end while**

14: **Send TLs in such a way that constraints (4.106) is satisfied with equality**
15: **while** $\hat{C}_{kl} > 0$ **do**
16: Let $B_{kl}^{mn} = \min\{B_{kl}^{ij} : \hat{y}_{kl}^{ij} < 1\}$ and $B_{lk}^{op} = \min\{B_{lk}^{ij} : \hat{y}_{lk}^{ij} < 1\}$;
17: **while** $(B_{kl}^{mn} + (1-\Theta) B_{lk}^{op}) < 0$ **do**
18: Set $f^{kl} = w_{mn}(1 - \hat{y}_{kl}^{mn})$;
19: **if** $f^{kl} > \frac{w_{op}(1-\hat{y}_{lk}^{op})}{(1-\Theta)}$ **then**
20: $f^{kl} = \frac{w_{op}(1-\hat{y}_{lk}^{op})}{(1-\Theta)}$;
21: **end if**
22: **if** $(2-\Theta) f^{kl} > \hat{C}_{kl}$ **then**
23: $f^{kl} = \frac{\hat{C}_{kl}}{(2-\Theta)}$;
24: **end if**
25: $f^{lk} = (1-\Theta) f^{kl}$;
26: $\hat{Z}_{\mathbf{y}}^{kl} = \hat{Z}_{\mathbf{y}}^{kl} + B_{kl}^{mn}\, f^{kl} + B_{kl}^{op}\, f^{lk}$; $\hat{C}_{kl} = \hat{C}_{kl} - w_{mn}\, f^{kl} - w_{op}\, f^{lk}$;
27: $\hat{y}_{kl}^{mn} += \frac{f^{kl}}{w_{mn}}$ and $\hat{y}_{lk}^{op} += \frac{f^{lk}}{w_{op}}$;
28: **end while**
29: **end while**

In Algorithm 5, $\hat{Z}_{\mathbf{y}}^{kl}$ is the objective function value from Algorithm 4. $\hat{C}_{kl}$ is the leftover capacity of the link (k,l) and B_{kl}^{ij} is the associated coefficient of y_{kl}^{ij}. In steps 5-

13, the algorithm removes flows in the direction (k,l) in descending order of B_{kl}^{ij} until constraint (4.106) is satisfied with equality; $\hat{C}_{kl}$ and y_{kl}^{ij} are adjusted accordingly. The implied link-imbalance level $\bar{\Theta}$ is then equal to $\frac{\sum_i \sum_j w_{ij}\, y_{kl}^{ij} - \sum_i \sum_j w_{ij}\, y_{lk}^{ij}}{\sum_i \sum_j w_{ij}\, y_{kl}^{ij}} = \Theta$. Steps 15-30 improve $\hat{Z}_{\mathbf{y}}^{kl}$ by re-sending flows in such a way that $\bar{\Theta}$ is maintained at Θ, and $(1-\Theta)$ units of flow are sent in the (l,k) direction whenever a unit of flow is sent in the (k,l) direction. For this, the algorithm first matches the best commodity, $[m,n]$ and $[o,p]$, in direction (k,l) and (l,k). If they satisfy the condition $(B_{kl}^{mn} + (1-\Theta)B_{lk}^{op}) < 0$, then augmenting these commodities can improve $\hat{Z}_{\mathbf{y}}^{kl}$.

The flows of the commodity $[m,n]$ are represented by f_{kl}. Initially, f_{kl} is equal to the whole demand w_{mn} if $\hat{y}_{kl}^{mn} = 0$ or equal to the leftover demand $w_{mn}(1-\hat{y}_{kl}^{mn})$ if $\hat{y}_{kl}^{mn} > 0$. In the opposite direction, $(1-\Theta)f_{kl}$ of the commodity $[o,p]$ must be sent. However, if there is not enough leftover demand ($f^{kl} > \frac{w_{op}(1-\hat{y}_{lk}^{op})}{(1-\Theta)}$), then f_{kl} is adjusted to $\frac{w_{op}(1-\hat{y}_{lk}^{op})}{(1-\Theta)}$. With respect to link capacity, these additional TLs must not exceed the available capacity $\hat{C}_{kl}$. If $f_{kl} + (1-\Theta)f_{kl} = (2-\Theta)f_{kl} > \hat{C}_{kl}$, then f_{kl} is re-adjusted to $\frac{\hat{C}_{kl}}{(2-\Theta)}$. Finally, after f_{kl} is determined, f_{lk} is readily set to $(1-\Theta)f_{kl}$ and $\hat{Z}_{\mathbf{y}}^{kl}$, $\hat{y}_{kl}$, and $\hat{C}_{kl}$ are adjusted according to the additional flows, f_{kl} and f_{lk}. The algorithm repeats these steps until the leftover capacity is used up or the augmentation condition, $(B_{kl}^{mn} + (1-\Theta)B_{lk}^{op}) < 0$, is not satisfied, i.e., additional flows only increase $\hat{Z}_{\mathbf{y}}^{kl}$.

To show that Algorithm 5 can construct an optimal solution to RRND_y^{kl}, we first note that Algorithm 5 provides the best y_{kl} that has $\bar{\Theta} = \Theta$. Thus, we only have to prove that there exists an optimal solution with $\bar{\Theta} = \Theta$. To do so, we define the following notation:

- Let $\hat{y}_{kl}$ be the solution of Algorithm 4. The flows in directions (k,l) and (l,k) are $a = \sum_i \sum_j w_{ij}\, \hat{y}_{kl}^{ij}$ and $b = \sum_i \sum_j w_{ij}\, \hat{y}_{lk}^{ij}$, respectively ($\frac{a-b}{a} > \Theta$).

- Let $\tilde{y}_{kl}$ be the solution of Algorithm 5. The flows in directions (k,l) and (l,k)

are $c = \sum_i \sum_j w_{ij}\, \tilde{y}_{kl}^{ij}$ and $e = \sum_i \sum_j w_{ij}\, \tilde{y}_{kl}^{ij}$. $\frac{c-d}{c} = \Theta$. Z_{kl}^{cd} is their associated objective function value.

- Let $\bar{y}_{kl}$ be the optimal solution. The optimal flows in direction (k,l) and (l,k) are e and f, respectively. Clearly, $f \geq b$, and Z_{kl}^{ef} is the optimal objective function value where $Z_{kl}^{ef} < Z_{kl}^{cd}$.

Clearly, e cannot be greater than a. If Algorithm 4 terminates because of exhausted capacity, then the level of e greater than a violates the capacity constraint. Alternatively, if f can be reduced below b to create extra space for e to increase beyond a, then the link-imbalance constraint will be violated. On the other hand, if the algorithm terminates when it cannot find negative B_{kl}^{mn}, then additional flows beyond a can only worsen the objective function value. Based on these observations, we can show that Algorithm 5 provides an optimal solution to RRND_y^{kl}.

Proposition 1. There exists an optimal solution with $\bar{\Theta} = \frac{c-d}{c} = \Theta$ and the objective function value of Z_{kl}^{cd}.

Proof. There are four possible cases to be considered.

1. $e < c$ and $f < d$: we can find two commodities [m,n] and [o,p] such that $(B_{kl}^{mn} + (1-\Theta)B_{lk}^{op}) < 0$ and their augmentations can improve Z_{kl}^{ef}.

2. $e > c$ and $f < d$: $\frac{e-f}{e} > \frac{c-d}{c} = \Theta$, this contradicts the assumption that the optimal flows in direction (k,l) and (l,k) are e and f.

3. $e < c$ and $f > d$: we let $\mathcal{R}$ be the set of augmented commodities in the (k,l) direction of $\tilde{y}_{kl}^{ij}$ but not in the optimal solution where $r_1, r_2, \ldots, r_R \in \mathcal{R}$ and $B_{kl}^{r_1} \leq B_{kl}^{r_2} \leq \ldots \leq B_{kl}^{r_R}$. Similarly, we let $\mathcal{S}$ be the set of augmented commodities the (l,k) direction of the optimal solution but not in $\tilde{y}_{lk}^{ij}$ where

$s_1, s_2, \ldots, s_S \in \mathcal{S}$ and $B_{lk}^{s_1} \leq B_{lk}^{s_2} \leq \ldots \leq B_{lk}^{s_S}$. Clearly, $B_{kl}^{r_R} \leq 0$, $B_{kl}^{r_R} \leq B_{lk}^{s_S}$, and $Z_{kl}^{ef} - Z_{kl}^{cd} = (B_{lk}^{s_1} w_{s_1} \bar{y}_{lk}^{s_1} + B_{lk}^{s_2} w_{s_2} \bar{y}_{lk}^{s_2} + \ldots + B_{lk}^{s_S} w_{s_S} \bar{y}_{lk}^{s_S}) - (B_{kl}^{r_1} w_{r_1} \tilde{y}_{kl}^{r_1} + B_{kl}^{r_2} w_{r_2} \tilde{y}_{kl}^{r_2} + \ldots + B_{kl}^{r_R} w_{r_R} \tilde{y}_{kl}^{r_R})$.

3.1) if $f - d \geq c - e$, then $(w_{r_1} \tilde{y}_{kl}^{r_1} + w_{r_2} \tilde{y}_{kl}^{r_2} + \ldots + w_{r_R} \tilde{y}_{kl}^{r_R}) \geq (w_{s_1} \bar{y}_{lk}^{s_1} + w_{s_2} \bar{y}_{lk}^{s_2} + \ldots + w_{s_S} \bar{y}_{lk}^{s_S})$, and $Z_{kl}^{fg} - Z_{kl}^{cd} \geq 0$.

3.2) If $f - d < c - e$, then the algorithm terminates because $(B_{kl}^{R_1} + (1-\Theta) B_{lk}^{S_1}) \geq 0$ and $B_{lk}^{s_1} \geq 0$. Therefore, $Z_{kl}^{ef} - Z_{kl}^{cd} \geq 0$.

4. $e > c$ and $f > d$: then $\left(\frac{(e-c)-(f-d)}{(e-c)}\right)$ cannot be greater than Θ in order to satisfy constraint (4.106). Thus, $(1-\Theta)(e-c)$ is less than or equal to $(f-d)$. We define $\mathcal{S}$ as in case 3 but redefine $\mathcal{R}$ as the set of augmented commodities in the (k,l) direction of the optimal solution but not in $\tilde{y}_{kl}^{ij}$. The other properties of $\mathcal{R}$ remain the same. In this case, the algorithm terminates because $(B_{kl}^{r_1} + (1-\Theta) B_{lk}^{s_1}) \geq 0$ and $B_{lk}^{s_1} \geq 0$. Therefore, $Z_{kl}^{fg} - Z_{kl}^{de} = (B_{kl}^{r_1} w_{r_1} \bar{y}_{kl}^{r_1} + \ldots + B_{kl}^{r_R} w_{r_R} \bar{y}_{kl}^{r_R} + B_{lk}^{s_1} w_{s_1} \bar{y}_{lk}^{s_1} + \ldots + B_{lk}^{s_S} w_{s_S} \bar{y}_{lk}^{s_S}) \geq B_{kl}^{r_1}(e-c) + B_{lk}^{s_1}(f-d) \geq B_{kl}^{r_1}(e-c) + (1-\Theta) B_{lk}^{s_1}(e-c) \geq (B_{kl}^{r_1} + (1-\Theta) B_{lk}^{s_1})(e-c) = 0$.

All four cases contradict the assumption that $Z_{kl}^{ef} < Z_{kl}^{cd}$; therefore, the solution obtained from Algorithm 5 is an optimal solution of RRND_y^{kl}.□

To better illustrate Algorithm 4 and Algorithm 5, we use Example 1 presented below for demonstration purposes.

Example 1. Consider the link $(1,2)$ where $c_{12} = 10$ and $\theta = 0.5$. B_{12}^{ij}, B_{21}^{ij}, and w_{ij} are provided in Table 16.

1. Algorithm 4 provides $y_{12}^{12} = y_{12}^{34} = y_{12}^{56} = y_{21}^{42} = 1$ and $y_{21}^{61} = 0.5$ where $\frac{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} + w_{56}\, y_{12}^{56} - w_{42}\, y_{21}^{42} - w_{61}\, y_{21}^{61})}{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} + w_{56}\, y_{12}^{56})} = 0.75 > 0.5$, $\hat{C}_{12} = 0$, and $Z = -76$.

Table 16: Example for Algorithm 5

$[i,j]$	B_{12}^{ij}	w_{ij}	$[i,j]$	B_{21}^{ij}	w_{ij}
$[1,2]$	-10	3	$[4,2]$	-5	1
$[3,4]$	-9	3	$[6,1]$	-4	2
$[5,6]$	-6	2	$[5,7]$	-1	1
$[7,8]$	-3	2	$[7,9]$	-1	2

2. Therefore Algorithm 5 must be utilized to correct this solution.

3. At step 13, the solution now becomes $y_{12}^{12} = 1$, $y_{12}^{34} = 0.33$, $y_{21}^{42} = 1$, and $y_{21}^{61} = 0.5$ where $\frac{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} - w_{42}\, y_{21}^{42} - w_{61}\, y_{21}^{61})}{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34})} = 0.5$, $\hat{C}_{12} = 4$ and $Z = -46$. Algorithm 5 then tries to improve the solution.

4. Since $(B_{12}^{34} + (1 - 0.5)B_{21}^{61}) = -11 < 0$, then Z can be improved by sending greater flow of commodities $[3,4]$ and $[6,1]$.

5. $f^{34} = 3 \times (1 - 0.33) = 2$ and $f^{61} = (1 - 0.5) \times f^{34} = 1$. Now the solution becomes $y_{12}^{12} = y_{12}^{34} = y_{21}^{42} = y_{21}^{61} = 1$ where $\frac{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} - w_{42}\, y_{21}^{42} - w_{61}\, y_{21}^{61})}{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34})} = 0.5$, $\hat{C}_{12} = 1$ and $Z = -70$.

6. Since $(B_{12}^{56} + (1 - 0.5)B_{21}^{57}) = -6.5 < 0$, then Z can be improved by sending greater flow of commodities $[5,6]$ and $[5,7]$.

7. $f^{56} = 2 \times (1 - 0) = 2$. However $(2 - \Theta)f^{56} > \hat{C}_{12}$, therefore $f^{56} = \frac{\hat{C}_{21}}{(2-\Theta)} = 0.67$ and $f^{57} = (1 - 0.5) \times f^{56} = 0.33$. Now the solution becomes $y_{12}^{12} = y_{12}^{34} = y_{21}^{42} = y_{21}^{61} = 1$, $y_{12}^{56} = y_{21}^{57} = 0.33$ where $\frac{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} + w_{56}\, y_{12}^{56} - w_{42}\, y_{21}^{42} - w_{61}\, y_{21}^{61} - w_{57}\, y_{21}^{57})}{(w_{12}\, y_{12}^{12} + w_{34}\, y_{12}^{34} + w_{56}\, y_{12}^{56})} = 0.5$, $\hat{C}_{12} = 0$ and $Z = -74.33$.□

8. Stop because $\hat{C}_{12} = 0$.

IV.2.3. Upper Bound Heuristic

The objective for our upper bound heuristic is to utilize solutions from RRND and quickly convert them into good feasible solutions. Note that the optimal solution is obtained if the lower bound solution is feasible. However, This is not usually the case since the flow conservation constraints are not included in RRND_y and $\mathbf{y}$ variables typically do not define a path. Thus, commodities must be re-routed in order to obtain their transmission paths.

The upper bound heuristic interprets from the $\mathbf{x}$ variables 1) the location of RPs and 2) the assignment of nonRP nodes, then utilizes this information to construct a distance matrix used in the commodity re-routing process. The re-routing process involves solving for the shortest path of each commodity. Each time after the shortest path is obtained, the flow is sent through the shortest path, and the leftover capacity of each link (along the shortest path) is updated accordingly. To avoid violating the capacity constraints, the transmission costs in both directions of a link (entries in the distance matrix) are set to a large number whenever the total flows on the link reach the capacity limitation; this makes the transmission over the exhausted links unfavorable.

For the link-imbalance requirement, the heuristic tries to maintain a "balanced" flow on every transmission link by increasing the transmission cost on the direction with more flow, and reducing the cost on the other side. In addition, if the flow amount leads to a link-imbalance violation on some links, the process then decreases the flow amount in order to reduce the violation and allow the cost adjustment to potentially find a feasible alternative path. With this dynamic cost/flow adjustment, every commodity is re-routed to facilitate finding a feasible solution and a valid upper bound. The re-routing process cannot guarantee the construction of feasible solutions.

Therefore, if an infeasible solution is found, the upper bound heuristic adjusts the distance matrix by 1) increasing the transmission cost on the direction of the link that induces the link-imbalance infeasibility, 2) reducing the cost on the other direction, and 3) decreasing the cost in both directions of feasible links. It then repeats the re-routing process in another attempt to find feasible solutions.

To illustrate the idea discussed above, we provide pseudo-codes for both the upper bound heuristic and the re-routing process in Algorithm 6 and Algorithm 7, respectively. For simplicity, we refer to the upper bound heuristic as Algorithm 6 and the re-routing process as Algorithm 7. Moreover, we define distance matrix D^1 with each entry $D^1_{kl} = d_{kl}$, $k, l \in \mathcal{N}$ if $d_{kl} \leq \Delta_2$ and $D^1_{kl} = M_2$ otherwise (M_2 is a large number).

Algorithm 6 Upper bound heuristic for Model 2

Set $Z_{UB} = M$, $G_{kl} = 0, D^2_{kl} = D^1_{kl}$, $k, l \in \mathcal{N}$;
for $i < n$ **do**
  Use algorithm in Algorithm 7 to obtain Z^i and $\mathbf{y}^i$;
  if $Z^i < Z_{UB}$ **then**
    $Z_{UB} = Z^i$ and $\mathbf{y} = \mathbf{y}^i$;
  end if
  if $\mathbf{y^i}$ is a feasible solution **then**
    STOP;
  else
    for $\forall (k,l), k, l \in RP$, $k < l$ that $G_{kl} + G_{lk} > 0$ **do**
      if $\frac{G_{kl} - G_{lk}}{G_{kl}} > \Theta$ **then**
        $D^2_{kl} = D^2_{kl} * \beta_1$ and $D^2_{lk} = D^2_{lk} * \alpha_1$;
      end if
      if $\frac{G_{lk} - G_{kl}}{G_{lk}} > \Theta$ **then**
        $D^2_{kl} = D^2_{kl} * \alpha_1$ and $D^2_{lk} = D^2_{lk} * \beta_1$;
      end if
      if $\frac{G_{kl} - G_{lk}}{G_{kl}} \leq \Theta$ and $\frac{G_{lk} - G_{kl}}{G_{lk}} \leq \Theta$ **then**
        $D^2_{kl} = D^2_{kl} * \alpha_1$ and $D^2_{lk} = D^2_{lk} * \alpha_1$;
      end if
    end for
    Set $G_{kl} = 0$;
  end if
end for

Algorithm 6 utilizes Algorithm 7 to construct a solution and the associated upper bound (step 3). If a feasible solution is returned, then Algorithm 6 terminates with a valid upper bound (steps 7-8). Otherwise, if the infeasible solution (marked by an unrealistically large value) is constructed, then Algorithm 6 adjusts and re-inputs D^2 to Algorithm 7 for another attempt at constructing a feasible solution. D^2 is adjusted based on the level of usage and the link-imbalance level. To do this, we define G_{kl} as the number of TLs transported between RP k and l, and modify D^2_{kl} that has $G_{kl} + G_{lk} > 0$ using the following rules:

1. If the implied link-imbalance is greater than the permissible link-imbalance, $\bar{\Theta} = \max\{\frac{G_{kl}-G_{lk}}{G_{kl}}, \frac{G_{lk}-G_{kl}}{G_{lk}}\} > \Theta$, then the distance of the direction that defines the link-imbalance is multiplied by α_1 ($0 \leq \alpha_1 \leq 1$) and the opposite direction is multiplied by β_1 ($\beta_1 \geq 1$) (steps 11-16).

2. If $\bar{\Theta} \leq \Theta$, then both D^2_{kl} and D^2_{lk} are multiplied by α_1 (steps 17-19).

The purpose of α_1 is to make one direction more favorable, while β_1 makes one direction less favorable. After the adjustment process, D^2 is re-input to Algorithm 7. Algorithm 6 repeats steps 3-22 for n iterations and reports the best infeasible solution (or a feasible solution, if found).

The order of commodities to be re-routed in Algorithm 7 is determined by the length of the shortest path using the following procedure. For commodity $[i, j]$, we let $r(i)$ be the RP to which the origin i is assigned and $r(j)$ be the RP to which the destination j is assigned. Next, we calculate, for every commodity, the shortest path length from $r(i)$ to $r(j)$ over the distance matrix D^2 and represent their values using L_{ij}. Finally, L_{ij} are sorted in descending order and used as a re-routing order in steps 2-3 of Algorithm 7.

In the re-routing process, Algorithm 7 finds the shortest path – of the commodity

$[\hat{i}, \hat{j}]$ over a distance matrix D^3 – through which some flow F_{ij} (number of TLs) will be sent (step 6). Initially, F_{ij} is the minimum between the leftover demand $W_{\hat{i}\hat{j}}$ and the leftover capacity of the links along the shortest path. However, if 1) there is a link (k, l) that is not previously used ($G_{kl} + G_{lk} = 0$), or 2) there are already too many flows on direction (k, l) ($\frac{G_{kl}-G_{lk}}{G_{kl}} > \Theta$), then only small fractions of demand, $0 \leq \varepsilon \leq 1$, are sent (steps 10-15). In addition, if direction (k, l) currently has $\Theta = \frac{G_{kl}-G_{lk}}{G_{kl}} \leq \Theta$, then F_{ij} of at most $\frac{G_{lk}}{1-\Theta} - G_{kl}$ are sent (steps 16-18). The objective of this partial augmentation is to avoid constructing a solution that violates the link-imbalance constraints by allowing the algorithm to adjust matrix D^3 and find alternative shortest paths. After obtaining F_{ij}, $Z^i, W_{\hat{i}\hat{j}}, G_{kl}, \mathbf{y}$, and C_{kl} are adjusted accordingly (steps 21, 23, and 25).

After each augmentation, the distance matrix D^3 is updated (steps 24-41). Whenever the link capacity C_{kl} is used up, the associated entries D^3_{kl} and D^3_{lk} are set to a large number M_2 so as to prevent more flow (step 27). While there is still some leftover capacity, Algorithm 7 adjusts D^3_{kl} according to the current level of $\bar{\Theta}$. If $G_{kl} = G_{lk}$, then $\bar{\Theta} = 0$ and we reset the associated D^3_{kl} and D^3_{lk} to D^2_{kl} and D^2_{lk}, respectively (steps 29-31). When $G_{kl} > G_{lk}$ and $\hat{\Theta} = \frac{G_{kl}-G_{lk}}{G_{kl}} \geq \Theta$, there are too many TLs on direction (k, l) and, therefore, D^3_{kl} is set to $D^2_{kl} * \beta_2$ ($\beta_2 \geq 1$) to make it less favorable and D^3_{lk} is set to $D^2_{lk} * \alpha_2$ ($\alpha_2 \leq 1$) to encourage more flow in this (l, k) direction (steps 33-34). If $\hat{\Theta} < \Theta$, then the algorithm sets $D^3_{lk} = D^2_{lk} * (1 - \hat{\Theta})$ and leaves D^3_{kl} with its current value (step 36). When $G_{kl} < G_{lk}$ and $\hat{\Theta} = \frac{G_{lk}-G_{kl}}{G_{lk}} \geq \Theta$, similar processes are applied with the exception that directions (k, l) and (l, k) must be reversed.

Algorithm 7 Re-routing process for Model 2

Set $Z^i = 0$ and $C_{kl} = c_{kl}$, $k, l \in \mathcal{N}, k < l$; Set $G_{kl} = 0, D^3_{kl} = D^2_{kl}$, $k, l \in \mathcal{N}$;
Sort L_{ij} in descending order and let $[\hat{i}, \hat{j}]$ be the sorted commodity indices;
for each sorted commodity $[\hat{i}, \hat{j}]$ **do**
  Let $W_{\hat{i}\hat{j}} = w_{\hat{i}\hat{j}}$;
  while $W_{\hat{i}\hat{j}} > 0$ **do**
    Solve the shortest path from $r(\hat{i})$ to $r(\hat{j})$ using the distance matrix D^3;
    Let $\bar{A}$ be the set of edges in the shortest path and $\bar{D}_{\hat{i}\hat{j}} = \sum_k \sum_l D_{kl}$, $(k,l) \in \bar{A}$;
    $F_{\hat{i}\hat{j}} = \min\{W_{\hat{i}\hat{j}}, \min\{C_{kl}, (k,l) \in \bar{A} \text{ s.t. } D^1_{kl} < M, C_{kl} > 0\}\}$;
    for $(k,l) \in \bar{A}$ **do**
      if $G_{kl} + G_{lk} = 0$ and $F_{\hat{i}\hat{j}} > \varepsilon$ **then**
        $F_{\hat{i}\hat{j}} = \varepsilon$;
      else
        if $\frac{G_{kl} - G_{lk}}{G_{kl}} > \Theta$ and $F_{\hat{i}\hat{j}} > \varepsilon$ **then**
          $F_{\hat{i}\hat{j}} = \varepsilon$;
        end if
        if $\frac{G_{kl} - G_{lk}}{G_{kl}} \leq \Theta$ and $F_{\hat{i}\hat{j}} > \frac{G_{lk}}{1-\Theta} - G_{kl}$ **then**
          $F_{\hat{i}\hat{j}} = \frac{G_{lk}}{1-\Theta} - G_{kl}$;
        end if
      end if
    end for
    $Z^i += (T_2 \times \bar{D}_{\hat{i}\hat{j}} \times F_{\hat{i}\hat{j}})$; $W_{\hat{i}\hat{j}} = W_{\hat{i}\hat{j}} - F_{\hat{i}\hat{j}}$;
    for $(k,l) \in \bar{A}$ **do**
      $G_{kl} += F_{\hat{i}\hat{j}}$; $\quad y^{\hat{i}\hat{j}}_{kl} += \frac{F_{\hat{i}\hat{j}}}{w_{\hat{i}\hat{j}}}$;
      if $D_{kl} < M_2$ **then**
        $C_{kl} = C_{kl} - F_{\hat{i}\hat{j}}$ (assume $k < l$, otherwise use $C_{lk} = C_{lk} - F_{\hat{i}\hat{j}}$);
        if $C_{kl} = 0$ **then**
          Set $D^3_{kl} = D^3_{lk} = M_2$;
        else
          if $G_{kl} = G_{lk}$ **then**
            Set $D^3_{kl} = D^2_{kl}$ and $D^3_{lk} = D^2_{lk}$;
          else
            if $G_{kl} > G_{lk}$ (for $G_{kl} < G_{lk}$, change every (k,l) to (l,k) and (l,k) to (k,l)) **then**
              if $\hat{\Theta} = \frac{G_{kl} - G_{lk}}{G_{kl}} \geq \Theta$ **then**
                $D^3_{kl} = D^2_{kl} * \beta_2$ and $D^3_{lk} = D^2_{lk} * \alpha_2$;
              else
                $D^3_{lk} = D^2_{lk} * (1 - \hat{\Theta})$;
              end if
            end if
          end if
        end if
      end if
    end for
  end while
end for
$\mathbf{y}^i = y^{\hat{i}\hat{j}}_{kl}$, $k, l \in \mathcal{N}$;
if $\mathbf{y}^i$ violate the capacity or link-imbalance constraints **then**
  $Z^i = Z^i \times 10$;
end if

Algorithm 7 repeats the re-routing process in steps 9-46 for every commodity and returns **y** variables with the associated objective function value and the matrix G to Algorithm 6.

In our preliminary experiments, we calibrate the algorithms' parameters in order to obtain good algorithmic performance and observe that the combination of $\alpha_1 = 0.01$, $\beta_1 = 1$, and n = 10 (in Algorithm 6), and $\alpha_2 = 0.1$ and $\beta_2 = 5$ (in Algorithm 7) usually provide good results.

IV.2.4. Overall Framework

The overall framework of our LD algorithm is presented in Algorithm 8. Initially, the best lower bound, LB_{best}, is set to 0 and the best upper bound, UB_{best}, is set to a large number M_3. The number of iterations $Iter$ and the number of consecutive iterations that the best lower bound is not improved t_{ni} are also initialized to 0. Since the set of Lagrangean multipliers is updated using UB_{best} in the subgradient optimization (presented in Section IV.2.2.2), we consider initializing M_3 to a realistic value. To do so, we assume that every node is an RP and use the upper bound heuristic to obtain the initial upper bound. We found that this procedure usually provides a feasible solution and a meaningful M_3.

Algorithm 8 Pseudo-code for Lagrangean Decomposition

1: Set $LB_{best} = 0$, $UB_{best} = M_3$, $Iter = t_{ni} = 0$;
2: **while** $f^t > \varepsilon_f$ **do**
3: $Iter = Iter+1$; $t_{ni} = t_{ni} + 1$;
4: **Solve the lower bound problem for** Z^t_{LB};
5: **if** $Z^t_{LB} > LB_{best}$ **then**
6: $LB_{best} = Z^t_{LB}$
7: $t_{ni} = 0$
8: **end if**
9: **if** $t_{ni} = ni$ **then**
10: $f^t = f^t \times m^f$;
11: $t_{ni} = 0$
12: **end if**
13: **Solve the upper bound problem for** Z^t_{UB};
14: **if** $Z^t_{UB} < UB_{best}$ **then**
15: $UB_{best} = Z^t_{UB}$
16: **end if**
17: **if** $(\frac{UB_{best}-LB_{best}}{UB_{best}} \leq \varepsilon_{opt})$ **then**
18: Stop;
19: **end if**
20: **Update the lagrangean multipliers using the subgradient optimization**;
21: **end while**

The algorithm first solves the RRND problem and obtains a valid lower bound Z^t_{LB}. Then, LB_{best} is replaced with Z^t_{LB} if $Z^t_{LB} > LB_{best}$ and t_{ni} is reset to 0 (steps 4-7). Note that if LB_{best} is not improved for ni consecutive iterations, then the factor f^t in the subgradient optimization is multiplied by m^f (step 10). Next, the algorithm utilizes the upper bound heuristic to construct an upper bound Z^t_{UB}. Similarly, UB_{best} is replaced by Z^t_{UB} if $Z^t_{UB} < UB_{best}$ (steps 13-15). Afterwards, the optimality gap $\frac{UB_{best}-LB_{best}}{UB_{best}}$ is calculated (steps 17-18) and if it is larger than the desired optimality gap ε_{opt}, the algorithm continue to update the Lagrangean multipliers using the subgradient optimization (step 20). The algorithm repeats the overall process until the optimality gap is small enough or when the the factor f^t is smaller than ε_f. In our

compensational experiments, we found that the combination of $f_t = 1.6$, $m^f = 0.4$, $\varepsilon_f = 0.0001$, and $ni = 20$ generally leads to a good algorithmic performance.

IV.2.5. Subgradient Method

The values of $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ have significant impacts on the convergence rate of our LD algorithm. In this section, we discuss how to utilize the subgradient optimization to obtain good candidates for $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$. In each iteration, after the optimality gap fails to terminate the algorithm, the following procedure is applied.

1. Let $\mathbf{x}$, $\mathbf{y}$, $\mathbf{u}$, and $\mathbf{v}$ be the solutions from RRND in the current iteration, then the $SSE = \sum_i \sum_j \sum_k (\sum_l y_{kl}^{ij} - u_k^{ij})^2 + \sum_i \sum_j \sum_k (\sum_l y_{lk}^{ij} - v_k^{ij})^2$.
2. Let t represent the iteration number, and let step size $s^t = f^t \frac{(UB_{best} - Z_{LB}^t)}{SSE}$.
3. Finally, we set $\sigma_k^{(t+1)ij} = \sigma_k^{tij} + s^t(\sum_l y_{kl}^{ij} - u_k^{ij})$ and $\tau_k^{(t+1)ij} = \tau_k^{tij} + s^t(\sum_l y_{lk}^{ij} - v_k^{ij})$.

The above procedure is applied for updating $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ in each iteration. However, for the initial $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ in the first iteration, we found that the following procedure provides a good starting point and helps with the convergence rate. First, the all-pair shortest paths, from one node to every other node, are solved and g_k^i is defined as the length of the shortest path from node i to node k. Then, for a commodity $[i, j]$, we set σ_k^{0ij} to $\frac{w_{ij}(g_k^i - g_k^j)}{2}$ and set τ_k^{0ij} to $\frac{w_{ij}(g_k^j - g_k^i)}{2}$.

IV.2.6. Computational Experiments

The objective of our computational experiments is to illustrate the efficiency of our LD algorithm and also to examine the influence of various parameters on both the algorithmic performance and the solutions characteristics. To solve the RPND$_{xyz}$, RPND$_{xy}$, and RRND$_{xuv}$ problems with the Branch-and-cut approach, we use CPLEX

9.1 with default settings for cut generation, preprocessing, and upper bound heuristics.

In the first experiment, we compare the performance of our LD algorithm to the Branch-and-cut (BC) approach as implemented in CPLEX. The first column in Table 17 contains the problem classes Ua3-4, Ub3-4, Uc3-4, and Ud3-4 with various combinations of $|\mathcal{N}|$ and D where $|\mathcal{N}|$ ranges from 20 to 40 nodes and D is both 60 and 80 percent. The Δ_1-Δ_2 value is fixed at 20-40 and the link capacity c_{kl} is assumed to have equal capacity c which is predetermined and set as presented in Table 17.

Table 17: Results of the BC and LD approaches (averages of 10 instances)

Problem class	B & C		LD ($Iter = 300$)				LD (3%)			
($\|\mathcal{N}\|$-D-c)	Time^z	Time^{xy}	Gap^{LD}	Gap^{lb}	Gap^{ub}	Time^{LD}	Gap ^{LD}	Gap^{lb}	Gap^{ub}	Time^{LD}
Ua3 (20-60-1300)	23	26	1.28	0.27	1.03	98	2.81	1.72	1.12	19
Ua4 (20-80-1300)	19	25	1.27	0.17	1.11	106	2.77[1]	1.76[2]	1.19[3]	28[4]
Ub3 (25-60-1500)	150	178	1.13	0.16	0.98	165	2.89	1.75	1.17	41
Ub4 (25-80-1500)	190	262	1.13	0.17	0.97	181	2.82	1.70	1.15	35
Uc3 (30-60-1700)	706	739	0.87	0.09	0.78	257	2.85	1.82	1.06	38
Uc4 (30-80-1700)	503	520	0.92	0.07	0.86	292	2.80	1.82	1.01	36
Ud3 (40-60-2000)	n/s	n/s	1.23	n/s	n/s	675	2.82	n/s	n/s	209
Ud4 (40-80-2000)	n/s	n/s	1.25	n/s	n/s	768	2.73	n/s	n/s	170

[1] The average of 9 instances without the outlier is 2.71%.
[2] The average of 9 instances without the outlier is 1.61%.
[3] The average of 9 instances without the outlier is 0.89%.
[3] The average of 9 instances without the outlier is 18 seconds.

In Table 17, Time^z and Time^{xy} are the runtimes that CPLEX takes to solve the formulations RPND$_{xyz}$ and RPND$_{xy}$ to optimality. It is clear that the BC approach is very effective in solving small instances and, due to the more compact and tighter formulation, RPND$_{xyz}$ is solved in slightly shorter runtimes. However, the runtimes and memory requirements for both formulations increase very rapidly with the increasing $|\mathcal{D}|$ and $|\mathcal{Q}|$. Eventually, the memory requirement grows prohibitively large and no solution can be obtained for classes Ud3 and Ud4.

Unlike the BC approach that employs a large amount of memory to exploit the

branch-and-bound tree, our LD approach requires significantly less memory by decomposing the problem and solving the parts separately. To illustrate the performance of our LD approach, we apply the LD algorithm to these problem classes and allow it to run for 300 iterations. The solution statistics are summarized in columns 4-7. We use Gap^{LD} and Time^{LD} to represent the LD optimality gap between the best lower and upper bounds at termination and the associated runtimes, respectively. Gap^{lb} is the percentage difference between the LD best lower bound, LB_{best}, and the optimal solution obtained from CPLEX. Likewise, Gap^{ub} is the percentage difference between the LD best upper bound, UB_{best}, and the optimal solution.

Clearly, the LD algorithm can efficiently solve all instances for the average optimality gap below 1.3% and, most importantly, classes Ud3 and Ud4, which the BC approach cannot solve, are now solved in less than 13 minutes. On average, the best lower bounds are less than 0.3% from optimality while the best upper bounds are approximately 1% greater than the optimal solution. In terms of runtimes, even though the LD algorithm takes longer to solve the small instances (Ua3-4), the runtimes grow at a slower rate when compared to the BC approach.

Although the number of iterations in the first experiment is set to 300, the best upper bound solutions are normally found in earlier iterations. For illustration purposes, we set the optimality gap ε_{opt} to 3%, re-solve the instances, and report the solution statistics in columns 8-11. In this setting, the LD approach can find good solutions verifiably within 3% optimal and which terminate at the early stages; the runtimes are relatively smaller than the 300 iterations criteria. Specifically, the best upper bounds are below 1.3% on average and the best lower bounds are below 2%. This indicates that the majority of the runtimes after the 3% gap is reached are spent improving the lower bounds as there is only a slight improvement in terms of the upper bound solutions.

Upon observing this tail-off effect, we decide to use 1) 300 LD iterations and 2) 3% optimality gap as the termination criteria for the rest of this experimentation. We also note that there is one instance in problem class C2 that the LD algorithm cannot solve to 3% optimality in 300 iterations. The lower bound gap is reported as small as 0.24% but the upper bound gap is 3.12%. In this case, the average gaps and runtimes without the outlier instance are reported.

In the second experiment, we solve the problem class Ud3-4, Ue3-4, and Uf3-4, each with 10 instances, using different settings of Δ_1, Δ_2, and c. The averages over 10 instances are reported in Table 18 where Iter is the number of LD iterations, T^{lb} and T^{ub} are the total time spent for solving for lower and upper bounds, #RP and #Link are the number of RPs and utilized links in the best upper bound solutions, $\bar{c}^A$ and $\bar{c}^M$ are the average and the maximum capacity usages, and $\bar{\Theta}^A$ and $\bar{\Theta}^M$ are the average and the maximum level of the implied link-imbalance. In all settings, the capacity and link-imbalance constraints are effective and we observe the tight values of $\bar{c}^M$ and $\bar{\Theta}^M$ that are very close or equal to c and Θ. We note that, even though $\bar{\Theta}^M$ is 0.41-0.54 (corresponding to 20.5-27% lane drivers' empty travel mileage), the value of $\bar{\Theta}^A$ is only 0.8-1.5 (4-7.5% empty miles). Clearly, the use of an RP-network can help control the empty back haul to a low level. Moreover, further improvement can be obtained if more flexible routing routines between multiple RPs are allowed (e.g., drivers can visit more than one RP in one trip – route RP_1-RP_2-RP_3-RP_1 becomes available in addition to RP_1-RP_2-RP_1).

To observe the impacts of modeling parameters, we abbreviate each Δ_1-Δ_2-c combination with a letter "a"-"g" as shown in column 2. With different Δ_1-Δ_2-c, the underlying structure of the RP-network is altered and different impacts on the algorithmic performance and solution characteristics can be observed. We examine the case with a different Δ_1 by comparing setting "c" to "e". When Δ_1 increases,

Table 18: Results of the LD approach with varying Δ_1-Δ_2-c values

Class ($\|\mathcal{N}\|$-D)	Δ_1-Δ_2-c	Ave value										
		Gap	Iter	Time	T^{lb}	T^{ub}	#RP	#Link	$\bar{c}^A$	$\bar{c}^M$	$\bar{\Theta}^A$	$\bar{\Theta}^M$
	20-40-2000 (a)	2.89	50	103	66	29	21	59	620	1841	0.11	0.46
Ud3	20-50-1000 (b)	2.95	89	196	134	44	20	72	422	1000	0.13	0.51
(40-60)	20-50-1500 (c)	2.92	46	96	68	18	20	67	450	1414	0.12	0.49
	20-50-2000 (d)	2.84	54	112	80	20	19	64	471	1749	0.12	0.49
	30-50-1500 (e)	2.93	66	148	112	22	16	55	499	1469	0.13	0.47
	30-60-1000 (f)	2.78	76	187	138	29	16	63	397	978	0.14	0.54
	30-60-1500 (g)	2.90	64	150	114	19	15	55	446	1450	0.13	0.52
	20-40-2000 (a)	2.74	42	103	69	26	22	62	637	1882	0.11	0.46
Ud4	20-50-1000 (b)	2.87	69	186	128	41	22	85	400	1000	0.13	0.54
(40-80)	20-50-1500 (c)	2.76	58	145	105	26	21	78	436	1394	0.13	0.49
	20-50-2000 (d)	2.80	66	167	124	28	21	78	434	1522	0.13	0.53
	30-50-1500 (e)	2.93	50	135	103	20	17	62	492	1469	0.13	0.52
	30-60-1000 (f)	2.95	64	193	144	30	18	82	346	997	0.14	0.53
	30-60-1500 (g)	2.81	82	238	181	32	17	73	376	1371	0.15	0.54
	20-40-4000 (a)	2.93	35	298	221	43	26	89	948	3808	0.10	0.45
Ue3	20-50-2000 (b)	2.92	47	478	353	54	24	107	678	2000	0.11	0.49
(60-60)	20-50-2500 (c)	2.96	38	385	285	42	24	101	706	2450	0.11	0.51
	20-50-4000 (d)	2.87	79	780	576	78	24	100	716	3194	0.11	0.50
	30-50-2500 (e)	2.92	51	611	481	49	21	93	735	2477	0.12	0.50
	30-60-2000 (f)	2.89	56	749	585	54	20	101	595	2000	0.13	0.53
	30-60-2500 (g)	2.92	50	638	491	48	19	94	628	2291	0.12	0.51
	20-40-4000 (a)	2.76	57	573	439	69	27	97	951	3852	0.09	0.42
Ue4	20-50-2000 (b)	2.90	51	633	482	62	26	122	654	2000	0.11	0.48
(60-80)	20-50-2500 (c)	2.87	55	686	522	64	25	119	661	2474	0.11	0.51
	20-50-4000 (d)	2.92	58	698	530	65	26	125	642	3227	0.10	0.51
	30-50-2500 (e)	2.85	64	903	716	72	23	105	719	2442	0.10	0.47
	30-60-2000 (f)	2.84	56	913	721	64	21	116	569	1977	0.11	0.53
	30-60-2500 (g)	2.85	86	1304	1014	99	22	118	572	2262	0.12	0.53
	20-40-5000 (a)	2.94	51	1246	879	151	32	125	1271	4984	0.09	0.49
Uf3	20-50-3000 (b)	2.90	51	1543	1094	143	29	150	876	3000	0.10	0.47
(80-60)	20-50-3500 (c)	2.96	44	1335	937	132	29	149	882	3362	0.10	0.50
	20-50-5000 (d)	2.95	40	1201	825	132	29	148	881	4007	0.10	0.52
	30-50-3500 (e)	2.95	50	1871	1442	123	27	141	894	3447	0.09	0.51
	30-60-3000 (f)	2.95	50	2075	1575	120	25	156	731	2827	0.10	0.49
	30-60-3500 (g)	2.94	46	1813	1340	124	23	130	846	3304	0.10	0.47
	20-40-5000 (a)	2.95	31	926	661	113	34	148	1186	4863	0.08	0.41
Uf4	20-50-3000 (b)	2.94	43	1609	1173	140	33	190	771	3000	0.10	0.52
(80-80)	20-50-3500 (c)	2.93	41	1505	1078	149	32	183	804	3440	0.09	0.49
	20-50-5000 (d)	2.92	46	1692	1201	175	31	167	885	4227	0.09	0.48
	30-50-3500 (e)	2.95	41	1733	1340	114	30	173	837	3487	0.09	0.47
	30-60-3000 (f)	2.88	45	2172	1649	133	26	164	759	2969	0.10	0.51
	30-60-3500 (g)	2.92	45	2159	1609	158	25	156	807	3413	0.10	0.53

the nonRP nodes can be assigned to an RP that is further away, thus fewer RPs are located and fewer links are available. As a result of fewer links, usages increase on the RP-RP links, as indicated by $\bar{c}^M$ and $\bar{\Theta}^M$.

On the other hand, comparing "a" to "d" and "e" to "g" illustrates the case when Δ_2 is increased. With a larger Δ_2, more RP-RP links satisfy the distance constraints, since truck drivers can now travel further in the RP-network. In this case, even though fewer RPs are located, the increased number of links is reported. These additional links improve network connectivity, provide alternative routes for heavily used links, and consequently promote a better distribution of link usages (lowered link capacity utilization). In addition, the alternative path (from the increased Δ_2) can help reduce the link-imbalance and, consequently, reduce the empty travel distance. To observe the impact of c, we compare the settings "b"-"d" and "f"-"g." When c increases, more flows are permitted between an RP-RP pair and higher link utilization is reported. This, in turn, leads to a reduced number of links being used in the RP-network.

In terms of algorithmic performance, increased Δ_1 and Δ_2 enlarge the solution space, thus the LD algorithm requires more iterations and runtimes. We note that the majority (around 74%) of the runtimes contribute to solving the lower bound problems while the other 12% are for solving the upper bound problem; the leftover 14% computational effort is spent on miscellaneous calculations (e.g., Lagrangean multipliers update). Although the T^{ub} increases as the problem class moves from Ud3 to Uf4, its percentage of the total runtimes decreases. In particular, the upper bound heuristics take less than 3 minutes to efficiently construct near optimal solutions (less than 3%) in all cases. Finally, we also note that the impact of c on the LD algorithm is not clearly illustrated in this experiment.

In the third experiment, we examine the influence of the node and commodity distributions. In this case, we generate the classes Uf3, Ug3, Uh3, and Ui3

and their clustered counterparts Cf3, Cg3, Ch3, and Ci3, each with 10 instances. Note that all these classes have $|\mathcal{N}| = 80$ and $D = 20$. Different distributions of nodes and commodities can significantly affect both the network configurations and the commodities' routes, especially in terms of link capacity requirements. To better illustrate their effects, we solve the uncapacitated setting (without the capacity constraints) and summarize the solution statistics in Table 19. We note that, in this experiment, the Δ_1-Δ_2 is kept constant at 20-40.

Table 19: Results from the uncapacitated model

Class	D-dist	$\bar{c}^M$	c^M	c
Uf3	60-20-20	5017	7622	4000
Ug3	20-60-20	3562	4813	2500
Uh3	20-20-60	2566	3073	1500
Ui3	40-30-30	4189	6502	3000
Cf3	60-20-20	11846	22979	10000
Cg3	20-60-20	7134	12015	6000
Ch3	20-20-60	5961	11283	4500
Ci3	40-30-30	8314	18843	8000

In Table 19, $\bar{c}^M$ is the average of the maximum link utilization over 10 instances, whereas c^M is the maximum of the link utilization level in all 10 instances. We can observe that $\bar{c}^M$ and c^M drop rapidly with the increased number of shorter commodities. In short (and some medium) range commodities, the origins and the destinations are usually located in the same cluster/region or in two nearby clusters/regions. Thus, fewer commodities travel between regions, which leads to a lowered link utilization. On the other hand, $\bar{c}^M$ and c^M increase more than twice when the nodes are located in clusters instead of uniformly distributed. A limited number of paths is available for transportation between clusters and, consequently, their RP-RP links are used

very intensively. Upon observing the link utilization in the uncapacitated model, we set c values in the capacitated setting in such a way that they activate the capacity constraints as summarized in the fifth column ($c \leq \bar{c}^M \leq c^M$).

In the fourth experiment (see Tables 20 and 21), we solve the capacitated setting with c values fixed as in Table 19, and Δ_1-Δ_2 fixed at 20-40. As indicated in Table 20, all instances can be solved to 3% optimality with the average runtimes below 6000 seconds except for two instances, one in the Ch3 setting and the other in the Ci3 setting. Both instances take approximately 13000 seconds to obtain the 3% optimality gap. This is due to very tight capacity constraints, as both of them are the two instances that pose the maximum uncapacitated c^M values of 11283 and 18843. Since the capacity limitations for c are set to 4500 and 8000, good feasible solutions are very difficult to find for these two instances. Thus, the average runtimes for these two settings are the average of 9 instances (without the outliers) and are indicated using italic numbers.

Table 20: Results from different node and commodity distributions

Uniform Node Distribution (Uf3-Ui3)								
Class	D-dist	Time	RP	Link	$\bar{c}^A$	$\bar{c}^M$	$\bar{\Theta}^A$	$\bar{\Theta}^M$
Uf3	60-20-20	1286	33	139	1127	3781	0.08	0.43
Ug3	20-60-20	1553	33	150	886	2469	0.07	0.41
Uh3	20-20-60	4502	38	193	596	1500	0.08	0.37
Ui3	40-30-30	1565	34	154	962	3000	0.08	0.40
Clustered Node Distribution (Cf3-Ci3)								
Class	D-dist	Time	RP	Link	$\bar{c}^A$	$\bar{c}^M$	$\bar{\Theta}^A$	$\bar{\Theta}^M$
Cf3	60-20-20	2007	26	101	1567	8785	0.09	0.49
Cg3	20-60-20	2240	26	96	1269	5534	0.08	0.42
Ch3	20-20-60	*5334*[1]	29	121	909	4079	0.08	0.35
Ci3	40-30-30	*1947*[1]	27	110	1359	7229	0.09	0.46

[1] The average of 9 instances.

With a greater number of shorter range commodities, excessive transportation

cost can be avoided by locating additional RPs. This consequently provides the RP-network with more connectivity and allows the shipments to travel on their best possible paths without having to strive for limited capacity on links that are shared by many commodities. As a result, link utilizations are distributed more evenly and both the capacity usage and link-imbalance levels are reduced.

Table 21: Comparing RP-network with direct shipments

Uniform Node Distribution (Uf3-Ui3)								
Class	D-dist	L^A	L^M	$\bar{d}^A$	$\bar{d}^M$	d^A	$\bar{\Omega}^A$	$\bar{\Omega}^M$
Uf3	60-20-20	4.0	8.6	87	186	81	0.13	11.57
Ug3	20-60-20	3.5	8.4	72	193	66	0.14	10.22
Uh3	20-20-60	3.1	9.0	60	191	53	0.19	14.41
Ui3	40-30-30	3.7	9.0	77	190	71	0.15	10.01
Clustered Node Distribution (Cf3-Ci3)								
Class	D-dist	L^A	L^M	$\bar{d}^A$	$\bar{d}^M$	d^A	$\bar{\Omega}^A$	$\bar{\Omega}^M$
Cf3	60-20-20	4.1	8.4	82	172	75	0.17	13.5
Cg3	20-60-20	3.5	8.3	67	178	59	0.21	16.3
Ch3	20-20-60	3.2	8.9	55	182	48	0.26	17.5
Ci3	40-30-30	3.8	8.8	73	181	65	0.22	18.4

In Table 21, L^A and L^M are the average and maximum number of legs per shipment. $\bar{d}^A$ and $\bar{d}^M$ are the average and maximum shipment distances from utilizing the RP-network. By assuming D^{ij} as the actual shipment distance of the commodity $[i, j]$ over the RP-network, we obtain the percentage of additional distance or percentage circuitry $\Omega_{ij} = \frac{D^{ij} - d_{ij}}{d_{ij}}$ where d_{ij} is the Euclidean distance between nodes i and j. Likewise, d^A is the average Euclidean distance of all the commodities. The average and the maximum percentage circuitry levels, calculated from Ω_{ij}, are also reported in Table 21 as $\bar{\Omega}^A$ and $\bar{\Omega}^M$, respectively.

The distances between the origins and destinations are shorter in short and medium range commodities, and they are relayed fewer times when compared to

the longer range ones (indicated by L^A and L^M). $\frac{\bar{d}^A}{L^A}$ is the average distance per leg and its values are around 20 in all cases. On the other hand, d^A can be used as the average distance per trip in PtP dispatching, if drivers are assumed to return to the RP from which they are dispatched after making direct shipments. Comparing $\frac{\bar{d}^A}{L^A}$ with d^A, we observe that the RP-network can reduce tour length significantly, from 48-81 miles/trip to as low as 22 miles/trip or less. This reduction of tour length (per trip) may be even more pronounced as actual trip distances in PtP can be much longer than d^A when multiple shipments are assigned to truck drivers; recall that d^A is simply a lower bound of the PtP tour length, since it is the distance of a single direct shipment. We note that this comes with only 13-26% of additional distance on average (indicated by $\bar{\Omega}^A$). In terms of percentage circuity, it is interesting to see that $\bar{\Omega}^A$ and $\bar{\Omega}^M$ increase when moving from classes Uf3 to Ui3 and Cf3 to Ci3. This is due to the increased number of shorter range commodities for which the same amount of additional travel distance is reflected as a larger percentage of circuitry.

Comparing the uniform and clustered node distributions, we observe that fewer RPs are required in clustered instances in order to cover all the nodes in entire service regions and, hence, fewer RP-RP links are available and utilized. This, in turn, causes capacity usage to increase dramatically, especially for those used in inter-region transportation. For the other statistics, there is no significant difference other than a slight increase in the link-imbalance and percentage circuitry, and a slight decrease in the actual route and Euclidean distances.

IV.2.7. Concluding Remarks

In this chapter, we considered the design of relay networks that explicitly address drivers' tour lengths and, at the same time, facilitate the control of empty back haul and capacity limitations. These requirements are incorporated into Model 2 in the

form of distance constraints (to control tour lengths), link-imbalance constraints (to facilitate the chance of finding a load on the back haul), and capacity constraints. Specifically, we determine 1) the relay point locations, 2) the assignment of nodes, and 3) the actual transportation routes for each commodity, in such a way that the stated requirements are satisfied.

Similar to Model 1, the MIP formulation of Model 2 is also highly constrained and very large in size. Hence, solving the formulation (and also the preprocessed formulation) with typical Branch-and-cut approaches appears very ineffective. In order to solve this model efficiently, we systematically devise the Lagrangean decomposition framework (for obtaining tight lower bounds) and upper bound heuristics to design a solution algorithm that can effectively solve large instances. Our algorithm can solve all instances to a small optimality gap within a reasonable period of time, as illustrated in the computational studies. In addition, our computational studies allow us to examine the impact of various problem and model parameters on the algorithmic performance and solution characteristics.

CHAPTER V

RELAY NETWORK DESIGN FOR TELECOMMUNICATIONS

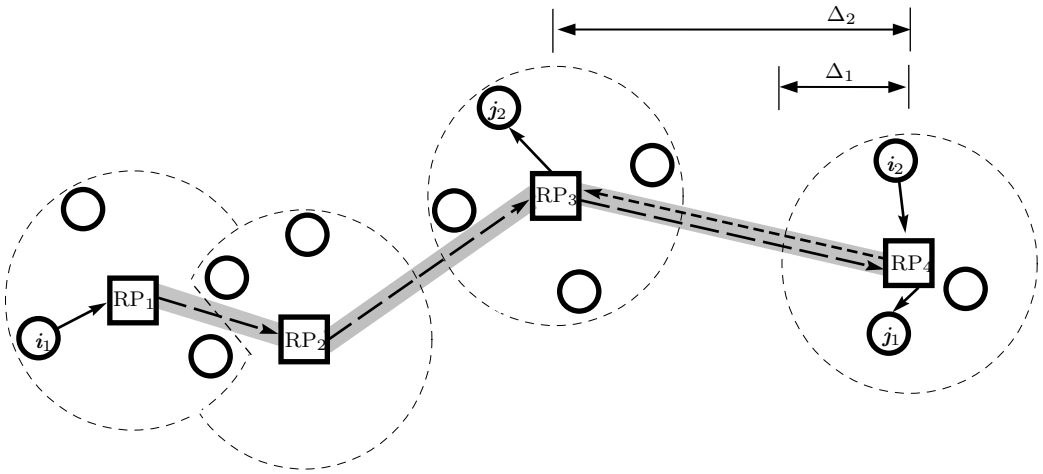
As discussed in Chapter I, telecommunications firms operate on physical networks to transmit signals and to connect end users who are scattered over a large service region. In long distance transmission, signals fade with distance and must be regenerated or strengthened by repeaters (relay points or RPs) in order to prevent the loss of signal and to reduce noise. In advance of transmitting signals and allowing communication between any two locations, a link connecting these locations must be established. However, due to the restriction of construction budgets, setting up links connecting all pairs of locations is not an option. Therefore, multiple users must share facilities (RPs and RP-RP links) in order to achieve cost effectiveness. For this purpose, a backbone network (RP-network) of RPs and RP-RP links must be constructed, with end users connecting to only a few, or even a single facility, within a proximity range. Employing this backbone RP-network, users connect to each other via communication channels formed by sequences of RPs connected by RP-RP links.

In order to design the RP-network for the telecommunications application discussed above, Model 3 in Section V.1 extends the base model to consider the case when links connecting RP pairs must also be established beforehand and the associated fixed link set-up cost must be charged prior to permitting signal transmission between RPs. Next, in Section V.2, we present Model 4 that further generalizes Model 3 by including a capacity limitation of total transmissions on the established RP-RP links. Particularly, Model 3 is a special case of Model 4 when the link capacity is unlimited.

V.1. Model 3: RNDP with Fixed Link Set-up Cost

In this section, we consider the variant of the base RNDP model where the set-up of links connecting RPs is required prior to their usage. Besides the link set-up and the associated fixed cost, the operational characteristics of the RP-network in Model 3 are identical to the base model.

Figure 7: A Schematic View of Model 3



In Figure 7, each node represents an end user location and a potential location of an RP. Nodes are represented by solid-line circles in which some RPs are located (represented by squares). Signals can be transmitted in both directions between RPs k and l, $k, l \in \mathcal{N}$, only after the RP-RP link (k, l) is established and the associated fixed cost F_{kl} is paid (e.g., commodities $[i_1, j_1]$ and $[i_2, j_2]$ utilized the established link (RP$_3$,RP$_4$) in different directions). Due to the limited transmission range, we assume that signals can travel at most Δ_2 between RPs, and at most Δ_1 between nonRP nodes and RPs, without losing the connection. This directly implies that the RP-

RP links can be at most Δ_2 and the nonRP nodes can access the RP-network only through the RPs that are within Δ_1 distance. We also assume a single assignment stating that each nonRP node must be connected to only one RP. Note that links between nonRP nodes and RPs are not required prior to their assignment. This is because, under the single assignment assumption, the associated fixed link (node RP-RP) set-up cost, if considered, can be directly incorporated into the node assignment cost terms.

V.1.1. Model

Based on the parameters and the decision variables defined in Sections II.1 and II.2, Model 3 be formulated as follows:

$$\text{Min } Z = \sum_i \sum_k T_1 \, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2 \, d_{kl} \, w_{ij} \, y_{kl}^{ij}$$
$$+ \sum_k F_k \, x_{kk} + \sum_k \sum_l F_{kl} \, z_{kl} \tag{5.1}$$

subject to

$$d_{ik}\, x_{ik} \leq \Delta_1 \qquad \forall\, i,k \in \mathcal{N} \tag{5.2}$$

$$d_{kl}\, z_{kl} \leq \Delta_2 \qquad \forall\, i,k \in \mathcal{N} \tag{5.3}$$

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{5.4}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{5.5}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,k \in \mathcal{N} \tag{5.6}$$

$$z_{kl} \leq x_{kk} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{5.7}$$

$$z_{kl} \leq x_{ll} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{5.8}$$

$$y_{kl}^{ij} \leq z_{kl} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k,l \in \mathcal{N},\, k < l \tag{5.9}$$

$$y_{lk}^{ij} \leq z_{kl} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k,l \in \mathcal{N},\, k < l \tag{5.10}$$

$$x_{ik},\, z_{kl} \in \{0,1\},\, y_{kl}^{ij} \geq 0 \qquad \forall\, i,j,k,l \in \mathcal{N} \tag{5.11}$$

In the above formulation, the objective function comprises two main components, the total signal transmission cost and the RP-network construction cost. The first two terms in (5.1) correspond to the costs that arise when signals are routed between nonRP nodes and RPs, and between any two RPs. The last two terms in (5.1) are the RP location and the RP-RP link set-up cost, respectively. Constraints (5.2) and (5.3) are the distance constraints defined by signal transmission ranges. Constraints (5.2) allow nonRP nodes to access the RP-network only through the RP that is within Δ_1 range, whereas constraints (5.3) restrict the setting up of the link if it is longer than Δ_2. Constraints (5.4) are the flow conservation constraints that define the transmission path on the backbone RP-network for each commodity. Constraints (5.5) enforce the single assignment to every node. Non-RP nodes can access the RP-

network through one unique RP. On the other hand, whenever a node is selected as an RP location, it becomes part of the backbone network and is assigned to itself. Hence, all signals originating at this location can be transmitted directly through RP-RP links. Constraints (5.6)-(5.10) specify the structural requirements of the RP-network. Constraints (5.6) prohibit the direct transmission between a nonRP origin-destination pair without utilizing the RP-network. Moreover, these constraints also state that a node must access the RP-network only though an RP, and relaying signals between a nonRP node pair is not allowed. Constraints (5.7)-(5.8) locate RPs on both ends of a link, once it is established. Constraints (5.9)-(5.10) prevent the transmission between any two RPs unless they are connected by RP-RP links. Finally, constraints (5.11) state that $\mathbf{x}$ and $\mathbf{z}$ are binary while $\mathbf{y}$ variables are real numbers. Although $\mathbf{y}$ variables are not binary, Model 3 has the integrality property when the values of $\mathbf{x}$ and $\mathbf{z}$ are given. Because of constraints (5.9)-(5.10) and the uncapacitated structure of the RP-network (uncapacitated RPs and links), signals are generally transmitted through the shortest possible channel in the RP-network. This, in turn, causes $\mathbf{y}$ variables to have values of either zero or one.

Model 3 is identical to the base model if the last term in the objective function is disregarded. Moreover, the structural defining constraints (5.7)-(5.10) are actually the same as constraints (4.70)-(4.73) in Model 2; however, they are utilized under different objectives. The $\mathbf{z}$ variables in Model 3 indicate the existence of physical RP-RP links while $\mathbf{z}$ variables in Model 2 are utilized to obtain a compact formulation, thus constraints (4.70)-(4.73) can be restated with $\mathbf{y}$ variables (instead of $\mathbf{z}$).

V.1.2. Benders Decomposition Framework

Because the formulation of Model 3 is closely related to those of the base model and Model 1, solving it with the branch-and-cut approach also appears inefficient because

of the rapid growth rate of the problem size. Unlike the branch-and-cut approach that utilizes a large amount of memory to exploit the branch-and-bound tree of a large MIP model, Benders decomposition (BD) devises an indirect approach that facilitates better management of the memory requirement. By transforming the formulation into a master problem and a subproblem, Benders cuts are generated from the subproblem and auxiliary variables, as needed, to tighten the master problem. This BD framework has shown promising results in the base model (in our preliminary studies) and Model 1 (Section IV.1), especially when the subproblem is decomposable into smaller LP problems which can further minimize memory usage. Additionally, upon applying the BD framework to Model 3, many accelerating techniques are applicable to enhance the algorithmic performance of the BD framework.

In order to develop the BD-base algorithm for Model 3, we follow closely the same methodological exposition of the algorithmic development presented in Section IV.1. Whenever the values of the network construction variables ($\mathbf{x}$ and $\mathbf{z}$) are given, the formulation reduces to an LP subproblem that involves only $\mathbf{y}$ variables. The $\mathbf{y}$ subproblem is essentially an uncapacitated multicommodity network flow problem, which can be further decomposed and simply solved as a series of shortest path problems, one for each commodity on a given RP-network. The decomposable property of the subproblem permits the generation of Benders cuts in many forms in which the most promising results can be achieved via cut Type D4 – the Benders cuts defined for each commodity – as shown in Chapter IV.1.5. Due to its capability of providing fast convergence, the BD-based algorithms for Model 3 are based on the assumption that Benders cuts Type D4 are generated and incorporated into the master problem. Along with the disaggregation of Benders cuts, other accelerating techniques in Section IV.1 are applied and tested in order to evaluate their algorithmic benefit.

V.1.2.1. Benders Subproblem and its Dual

For given $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ variables, we state a subproblem $\mathrm{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ for each commodity $[i,j] \in \mathcal{Q}$ as follows:

$$\mathrm{Min}\, Z_{\mathrm{SP}_{ij}} = \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} \tag{5.12}$$

subject to

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = \hat{x}_{jk} - \hat{x}_{ik} \qquad \forall\, k \in \mathcal{N}, \forall\, [i,j] \in \mathcal{Q} \tag{5.13}$$

$$y_{kl}^{ij} \leq \hat{z}_{kl} \qquad \forall\, k,l \in \mathcal{N},\ k < l \tag{5.14}$$

$$y_{lk}^{ij} \leq \hat{z}_{kl} \qquad \forall\, k,l \in \mathcal{N},\ k < l \tag{5.15}$$

$$y_{kl}^{ij} \geq 0 \qquad \forall\, k,\ l \in \mathcal{N}, \forall\, [i,j] \in \mathcal{Q} \tag{5.16}$$

Then, by defining α_k^{ij}, σ_{kl}^{ij} and τ_{kl}^{ij} as the dual variables associated with (5.13), (5.14) and (5.15), the dual subproblem $\mathrm{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ for $[i,j] \in \mathcal{Q}$ can be stated as follows:

$$\mathrm{Max}\, Z_{\mathrm{DSP}_{ij}} = \sum_k (\hat{x}_{jk} - \hat{x}_{ik})\, \alpha_k^{ij} + \sum_k \sum_l \hat{z}_{kl}(\sigma_{kl}^{ij} + \tau_{lk}^{ij}) \tag{5.17}$$

subject to

$$\alpha_l^{ij} - \alpha_k^{ij} + \sigma_{kl}^{ij} \leq T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\ l \in \mathcal{N},\ k \leq l \tag{5.18}$$

$$\alpha_l^{ij} - \alpha_k^{ij} + \tau_{kl}^{ij} \leq T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\ l \in \mathcal{N},\ k \geq l \tag{5.19}$$

$$\sigma_{kl}^{ij},\ \tau_{kl}^{ij} \leq 0, \quad \alpha_k^{ij} \text{ unrestricted} \qquad \forall\, k,\ l \in \mathcal{N},\ k \neq l \tag{5.20}$$

Since the disaggregated cut Type D4 is assumed, each time after the dual subproblem is solved, the Benders cut defined for each commodity $[i,j] \in \mathcal{Q}$ can be generated using the values of dual variables $\hat{\boldsymbol{\alpha}}^{\boldsymbol{ij}}, \hat{\boldsymbol{\sigma}}^{\boldsymbol{ij}}, \hat{\boldsymbol{\tau}}^{\boldsymbol{ij}}$, and an auxiliary continu-

ous variable B_{ij} as follows:

$$B_{ij} \geq \Big(\sum_{k} (x_{jk} - x_{ik})\, \hat{\alpha}_k^{ij} \sum_{k} \sum_{l} \hat{z}_{kl} (\hat{\sigma}_{kl}^{ij} + \hat{\tau}_{lk}^{ij}) \Big) \tag{5.21}$$

In order to generate valid Benders cuts, the subproblem $\mathrm{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ must be feasible. That is, the RP-network from the master problem (the given $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ variables) must contain a feasible shortest path with special characteristics specified by the subproblem $\mathrm{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$. In the base model, such feasibility is ensured by 1) assigning a large distance to each arc (k, l) that is longer than Δ_2, and 2) removing the distance constraints (4.6). As a result, the RP-network from the master problem is always feasible and the infeasibility in the RP-network (violation of distance constraints (4.6)), if it exists, is identified by the shortest path with unrealistically long distance. A similar approach is applied to Model 1; however, when the percentage circuitry is considered, the infeasibility in the subproblem arising from violating the percentage circuitry constraints (4.10) cannot be controlled.

In this section, the construction of a feasible RP-network also involves the set-up of links connecting RP pairs. Unless the given $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ variables form a connected network of RPs and RP-RP links, this additional requirement leads to infeasibility issues and the Benders cuts based on the extreme ray must be generated. However, even though the infeasible RP-network is given, not all commodities are infeasible. This is another benefit of the cut disaggregation scheme (Section IV.1.3.2) that permits the generation of Benders cuts as long as the infeasible RP-network contains a valid shortest path for such a commodity.

For a commodity $[i, j] \in \mathcal{Q}$ that $\mathrm{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ is infeasible and $\mathrm{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ is unbounded, we generate the Benders cut (5.22) that is based on the extreme ray.

$$\Big(\sum_{k} (x_{jk} - x_{ik})\, \hat{\alpha}_k^{ij} \sum_{kl} \sum_{l} \hat{z}_{kl} (\hat{\sigma}_{kl}^{ij} + \hat{\tau}_{lk}^{ij}) \Big) \leq 0 \tag{5.22}$$

V.1.2.2. Benders Master Problem

Utilizing the dual variables $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, and $\hat{\boldsymbol{\tau}}$ from the subproblems, the master problem MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) of Model 3 can be stated as follows:

$$\text{Min } Z_{\text{MP}} = \sum_i \sum_k T_1\, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_k F_k\, x_{kk} + \sum_k \sum_l F_{kl} z_{kl} + \sum_i \sum_j B_{ij} \tag{5.23}$$

subject to

$$d_{ik}\, x_{ik} \leq \Delta_1 \qquad \forall\, i,\, k \in \mathcal{N} \tag{5.24}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{5.25}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,\, k \in \mathcal{N} \tag{5.26}$$

$$z_{kl} \leq x_{kk} \qquad \forall\, k, l \in \mathcal{N},\, k < l \tag{5.27}$$

$$z_{kl} \leq x_{ll} \qquad \forall\, k, l \in \mathcal{N},\, k < l \tag{5.28}$$

(constraints for the set of *BCuts*) (5.29)

$$x_{ik},\ z_{kl}\ \in \{0, 1\} \qquad \forall\, i, k, l \in \mathcal{N} \tag{5.30}$$

Since the disaggregate cut Type D4 is earlier assumed for the development of the BD-algorithm for Model 3, the term *BCuts* represents the Benders cuts generated for each commodity, either in the form of (5.21) or (5.22). B_{ij} in the objective function are the auxiliary continuous variables that relate the subproblem to the master problem. Thus, the term $\sum_i \sum_j B_{ij}$ contributes to the total signal transmission cost. We present the base BD algorithm for Model 3 in Algorithm 9, which is very similar to Algorithm 1 in Section IV.1.2.3, except for the terms *Bvars* and *SumBvars*, which are replaced by B_{ij} and $\sum_i \sum_j B_{ij}$, respectively.

Algorithm 9 Base BD Algorithm for Model 3

Initialize $UB = \infty$, $B_{ij} = 0$, $\hat{\boldsymbol{\alpha}} = \hat{\boldsymbol{\sigma}} = \hat{\boldsymbol{\tau}} = 0$ and $Iter = 0$; $MaxIter$;
Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP} and $\hat{\mathbf{x}}$. Set $LB = Z_{MP}$;
while $Iter \leq MaxIter$ **do**
  Solve DSP($\boldsymbol{\alpha}$, $\boldsymbol{\sigma}$, $\boldsymbol{\tau}|\hat{\mathbf{x}}$) for Z_{DSP}, $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$;
  $Iter = Iter + 1$;
  if $Z_{\text{MP}} - \sum_i \sum_j B_{ij} + Z_{\text{DSP}} < UB$ **then**
    $UB = Z_{\text{MP}} - \sum_i \sum_j B_{ij} + Z_{\text{DSP}}$; $\bar{\mathbf{x}} = \hat{\mathbf{x}}$;
  end if
  if $(UB - LB)/\ LB \leq \varepsilon$ **then**
    break;
  end if
  Generate *BCuts* with $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}},$ and $\hat{\boldsymbol{\tau}}$ and incorporate them into MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{\tau}}$);
  Solve MP($\mathbf{x}|\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\sigma}}$, $\hat{\boldsymbol{\tau}}$) for Z_{MP}, $\hat{\mathbf{x}}$, and *Bvars*. Set $LB = Z_{MP}$;
  if $(UB - LB)/\ LB \leq \varepsilon$ **then**
    break;
  end if
end while
Solve SP($\mathbf{y}|\bar{\mathbf{x}}$) to obtain $\bar{\mathbf{y}}$;
$(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is the best solution upon termination.

Note that when the ε-optimal approach is applied to the base-BD algorithm, constraint (5.31) is incorporated into the master problem and the BD-algorithm follows the same adjustments applied in Section IV.1.3.3 (Algorithm 2).

$$\sum_i \sum_k T_1\, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_k F_k\, x_{kk} + \sum_k \sum_l F_{kl} z_{kl} + \sum_i \sum_j B_{ij} \;\leq\; UB(1-\varepsilon) \tag{5.31}$$

V.1.3. Approaches for Accelerating the Base Algorithm

Models 1 and 3 share several similarities and, except for some differences in the master and subproblem formulations, they are solved following the same BD framework. Thus, the accelerating techniques applied to Model 1 are applicable to Model 3.

In the following sections, we discuss how these techniques are customized for best practices in solving Model 3.

V.1.3.1. Strengthening the Benders Cuts

Each time the master problem is solved, we can interpret the set of RP nodes ($\mathcal{N}_{\text{RP}}$) and the assignment of the nonRP nodes from the value of $\hat{\mathbf{x}}$ variables, and the construction of RP-RP links from the value of $\hat{\mathbf{z}}$ variables. Utilizing this information, we can generate the complete RP-induced network, G_{RP}, where the arc lengths are set to $(T_2\, d_{kl}\, w_{ij})$, $\forall\, k,\, l \in \mathcal{N}_{\text{RP}}$ if the arc (k, l) has the corresponding $\hat{z}_{kl}$ equal to 1; otherwise, they are set to an arbitrarily large value. For each commodity $[i, j]$, by defining $r(i)$ and $r(j)$ as the RPs to which the origin and the destination are assigned, the subproblem $\text{SP}_{ij}(\mathbf{y}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$ is then reduced to finding the shortest path problem from $r(i)$ to $r(j)$ over the network G_{RP}. By letting L_{ij} be the length of the shortest path from $r(i)$ to $r(j)$, the strong Benders cuts can be obtained by solving the following linear program.

$$\text{Max} \qquad \sum_{k \in \mathcal{A}_j} \alpha_k^{ij} - \sum_{k \in \mathcal{A}_i} \alpha_k^{ij} + \sum_k \sum_l (\sigma_{kl}^{ij} + \tau_{kl}^{ij}) \tag{5.32}$$

subject to

$$\alpha_{r(j)}^{ij} = L_{ij}, \qquad \alpha_{r(i)}^{ij} = 0 \tag{5.33}$$

$$\sum_k (\hat{x}_{jk} - \hat{x}_{ik})\alpha_k^{ij} + \sum_k \sum_l \hat{z}_{kl}(\sigma_{kl}^{ij} + \tau_{kl}^{ij}) = L_{ij} \tag{5.34}$$

$$\alpha_l^{ij} - \alpha_k^{ij} + \sigma_{kl}^{ij} \le T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N},\ k < l \tag{5.35}$$

$$\alpha_l^{ij} - \alpha_k^{ij} + \tau_{kl}^{ij} \le T_2\, d_{kl}\, w_{ij} \qquad \forall\, k,\, l \in \mathcal{N},\ k > l \tag{5.36}$$

$$\sigma_{kl}^{ij},\, \tau_{kl}^{ij} \le 0, \quad \alpha_k^{ij} \text{ unrestricted} \qquad \forall\, k,\, l \in \mathcal{N} \tag{5.37}$$

In the objective function (5.32), $\mathcal{A}_i$ and $\mathcal{A}_j$ are the set of nodes that are within Δ_1 distance of nodes i and j, respectively. Similar to the strong Benders cut in Section IV.1.3.1, constraints (5.33) set two variables equal to their optimal values in $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$. Constraints (5.34), (5.35), (5.36), and (5.37) validate the generated Benders cuts with respect to the $\text{DSP}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\tau}|\hat{\mathbf{x}}, \hat{\mathbf{z}})$.

V.1.3.2. Derivation of Surrogate Constraints

We observe that the master problem's runtimes grow with the increased number of Benders cuts accumulated in the master problem. In this case, we develop a set of surrogate constraints for the purpose of speeding up the master problem's runtimes and reducing the number of iterations.

$$x_{kl} + x_{lk} + z_{kl} \leq 1 \qquad \forall\, k, l \in \mathcal{N},\ k < l \tag{5.38}$$

$$\sum_{l<k} z_{lk} + \sum_{l>k} z_{lk} \geq x_{kk} \qquad \forall\, k,\ l \in \mathcal{N} \tag{5.39}$$

$$\sum_{k} \sum_{l} z_{kl} \geq \sum_{i} x_{ii} - 1 \qquad \forall\, i, k, l \in \mathcal{N},\ k < l \tag{5.40}$$

Constraints (5.38) state that when node k is assigned to node l, then node l cannot be assigned to node k and the link (k, l) cannot be established. Constraints (5.39) ensure that every RP in a connected RP-network must be connected with at least one RP-RP link. Constraints (5.40) are based on the fundamental network property stating that a tree on n nodes contains exactly n-1 arcs (Ahuja et al., 1993). Since circles are allowed in the construction of a connected RP-network, the number of links established must be at least one less than the number of RPs located.

V.1.3.3. A Heuristic Algorithm to Enhance the Upper Bound

It was shown in the third experiment of Section IV.1.5 that the upper bound heuristic can improve the convergence of the BD-algorithm, both with the base and the ε-optimal BD algorithms. This observation motivates the use of heuristic algorithm in Model 3, especially when the infeasibility of the master problem's solution becomes an issue. Note that, whenever an infeasible $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ are input to the subproblem, the associated upper bound UB is unrealistically large. In addition, some of the Benders cuts must be derived from extreme rays. Thus, applying the upper bound heuristic facilitates finding an improved UB and generating extreme point-based Benders cuts that can be strengthened.

Prior to developing our upper bound heuristic, we define an "opened link" as a link (k, l) s.t. $d_{kl} \leq \Delta_2$, $x_{kk} = x_{ll} = 1$, and $\hat{z}_{kl} = 1$. A "closed link" is similar to the opened link except that its corresponding $\hat{z}_{kl} = 0$. By devising the most recent MP solution, $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$ in the current iteration, we define a distance matrix D where the entry D_{kl} is equal to $T_2 \times d_{kl} \times w_{ij}$ if $\hat{z}_{kl} = 1$, and M if $\hat{z}_{kl} = 0$. Note that matrix D is defined following the same procedure as in Section V.1.3.1. Therefore, the transmission cost of a commodity $[i, j]$ over the RP-network is equal to the the shortest path length L_{ij}, calculated using the distance matrix D from $r(i)$ to $r(j)$. In this case, the objective function value Z_{UB} of the MP solution is $Z_{MP} - \sum_i \sum_j B_{ij} + \sum_i \sum_j L_{ij}$. Moreover, while solving for the shortest path, the utilization frequency of the closed links are recorded in a link-utilization matrix U. Finally, the algorithm removes the fixed link established cost from Z_{UB} if the link is opened but is not used. The procedure of obtaining the associated upper bound from the set of opened links discussed herein is summarized in Algorithm 10.

Algorithm 10 Upper bound procedure for Model 3: $UB(\cdot)$

1: Set $Z_{UB} = Z_{MP} - \sum_i \sum_j B_{ij}$, $U_{kl} = 0, k, l \in \mathcal{N}, k < l$;
2: **for** each commodity $[i, j]$ **do**
3: Solve the shortest path from $r(i)$ to $r(j)$ wrt distance matrix D_{kl};
4: Let $\bar{\mathcal{A}}$ be the set of arcs in the shortest path and $L_{ij} = \sum_k \sum_l D_{kl}$, $(k, l) \in \bar{\mathcal{A}}$;
5: $Z_{UB} = Z_{UB} + L_{ij}$;
6: **for** $(k, l) \in \bar{\mathcal{A}}$ s.t. $d_{kl} \leq \Delta_2$ **do**
7: **if** $k < l$ and $z_{kl} = 0$ **then**
8: $U_{kl} = U_{kl} + 1$;
9: **end if**
10: **if** $k > l$ and $z_{lk} = 0$ **then**
11: $U_{lk} = U_{lk} + 1$
12: **end if**
13: **end for**
14: **end for**
15: $Z_{UB} = Z_{UB} - \sum_k \sum_l (F_{kl} \times \max\{0, z_{kl} - U_{kl}\})$;

As illustrated in Algorithm 11, the upper bound procedure in Algorithm 10 is utilized in the development of our upper bound heuristic. We let $UB(\cdot)$ be the associated upper bound obtained by applying the upper bound procedure to a set of opened links. By letting S_o and S_c be the set of opened and closed links from the MP solution, we attempt to find improved solutions by opening some of the closed links based on their popularity. Such popularity is dictated by the matrix U that records the frequency of each closed link when it appears in the upper bound solution.

To determine which links to open, the closed links S_c are sorted in descending order of U_{kl} (step 5) and q most popular links are selected from S_c (step 6); q takes a random value between 5 and 10. However, if q is greater than $|S_c|$, then all the links in S_c are opened. The new set of opened links, $S_o \cup S_q$, are evaluated using the upper bound procedure in Algorithm 10. If the new set of opened links improves the upper bound, then S_o, S_c, and U are updated based on the solution from the upper bound procedure (step 9). However, if the new set of opened links fails to improve the upper bound, then they are removed from consideration (step 11) and the new

set of q closed links are opened. The heuristic continues the process of finding the improved set of opened links until it terminates when all closed links are tried and an improved solution cannot be found. The objective function value of the heuristic solution is represented by Z_{LS}. We note that, in the worst case, the heuristic fails to improve the MP solution; $Z_{LS} = Z_{MP} - \sum_i \sum_j B_{ij} + Z_{DSP}$ (equivalent to the upper bound from solving the dual subproblem) is reported.

Algorithm 11 Upper bound heuristic for Model 3: UBH

Let S_o and S_c be the set of the opened and closed link obtained from MP;
Let $UB(S_o)$ be the upper bound associated with S_o;
Let $\bar{S}_c = S_c$;
while $\bar{S}_c \neq \varnothing$ **do**
  Sort $\bar{S}_c$ in descending order of U_{kl};
  $q = \min\{|\bar{S}_c|, R\}$, R is a random number between 5-10;
  Let S_q be the first q closed links in the sorted closed links $\bar{S}_c$;
  if $UB(S_o \cup S_q) < UB(S_o)$ **then**
    Update S_o, S_c, and U_{kl};
  else
    $\bar{S}_c = \bar{S}_c \setminus S_q$;
  end if
end while
$Z_{LS} = UB(S_o)$;

The only objective of the upper bound heuristic in Section IV.1.3.4 is to improve the best upper bound UB. However, in Model 3, we also generate Benders cuts based on the heuristic solution, and include them in MP in addition to the regular Benders cuts. This practice is motivated by the infeasible MP solution that forces the subproblem to generate numerous unstrengthened Benders cuts derived from extreme rays, especially in early iterations. Thus, by applying an improved feasible solution from the upper bound heuristic to the subproblem, Benders cuts derived only from extreme points can be achieved.

V.1.4. Computational Experiments

The objective of our computational experiment is to compare the performance of our BD algorithm as opposed to the Branch-and-cut approach, and to evaluate the benefit of each accelerating technique developed in the previous section. Additionally, we also examine the algorithmic performance under various parameter settings in order to illustrate their efficiency. Recall that the strong Benders cuts and the disaggregated cut Type D4 are assumed throughout the computational studies. CPLEX 9.1 with default settings is used whenever we solve the master problem and the dual subproblem, as well as when solving the formulation of Model 3 for benchmarking.

In the first experiment, we compare the performance of the Branch-and-cut (BC) approach with the base and the ε-optimal BD Algorithms and summarize their results in Table 22. Problem classes Ua1, Ub1, Uc1, and Ud1, each with 10 instances, are generated and solved to optimality in this experiment. Upon applying the BC approach to the generated instances, we observe a rapid growth of runtimes with increasing $\mathcal{N}$. Specifically, when we move to Ue1 ($\mathcal{N} = 60$), the BC approach runs out of memory and instances in this class cannot be solved. On the other hand, applying the base and the ε-optimal BD algorithms provides satisfactory results in providing runtimes comparable to those of the BC approach. In fact, for the class Ud1, the BD-based algorithms provide significantly better runtimes that grow at a much slower rate than those in the BC case. Comparing columns 4-6 to 7-9 illustrates the significance of the surrogate constraints derived in Section V.1.3.2. Note that the number of iterations and the MP runtimes can be reduced in all cases, and around 21-47% of the runtimes can be saved through the use of these constraints. Due to this beneficial reduction of runtimes and number of iterations, we incorporate the surrogate constraints in MP for the rest of this experimentation.

Table 22: Comparing base and ε-optimal BD algorithms with BC approaches

Problem Class	$\lvert\mathcal{N}\rvert$–D	BC Ave Time	Base BD Algorithm						Time red. (%)
			without(5.18)-(5.20)			with (5.18)-(5.20)			
			Ave Time	Ave MP Time	Ave Iter	Ave Time	Ave MP Time	Ave Iter	
Ua1	20-20	1.0	4.3	1.2	10.9	3.0	0.8	7.3	30.29
Ub1	25-20	4.4	13.2	3.6	14.1	8.9	2.6	8.9	32.18
Uc1	30-20	35.5	48.4	21.9	16.9	32.2	13.4	11.4	33.53
Ud1	40-20	491.3	239.2	127.3	19.0	126.7	73.6	9.8	47.03
Problem Class	$\lvert\mathcal{N}\rvert$–D	BC Ave Time	ε-optimal BD Algorithm						Time red. (%)
			without(5.18)-(5.20)			with (5.18)-(5.20)			
			Ave Time	Ave MP Time	Ave Iter	Ave Time	Ave MP Time	Ave Iter	
Ua1	20-20	1.0	4.6	1.1	13.0	3.7	0.6	8.8	21.23
Ub1	25-20	4.4	13.7	2.4	16.6	9.9	1.8	10.9	27.87
Uc1	30-20	35.5	39.1	10.0	18.8	28.7	8.0	13.4	26.73
Ud1	40-20	491.3	275.6	125.0	25.2	153.0	69.5	15.2	44.47

In the second experiment, we illustrate the benefits of employing the upper bound heuristic in enhancing the performance of the Base and ε-optimal BD algorithms. In the previous experiment, we already observed good performance from utilizing the strong Benders cuts, disaggregate cut D4, and surrogate constraints. Thus, when the heuristic is also considered, larger problem classes Ud1-2, Ue1-2, and Uf1-2 ($\mathcal{N} = 40$, 60, and80, $D = 20$ and 40) are generated and solved to 2% optimality (to avoid tail-off effect). The results are summarized in Table 23. Columns 2 and 8 indicate different practices of the heuristic in the BD-algorithms: "x" represents the case when the heuristic is not utilized; "1" implies the use of the heuristic only to improve the UB; and "2" indicates the BD-algorithm that devises both the improved UB and the Benders cuts generated from the heuristic solution.

Table 23: BD and ε-optimal algorithms with different local searches

	Base BD Algorithm						ε-optimal BD Algorithm					
Class	LS	Ave Time	Ave red. %	MP Time	MP Iter	SP Iter	LS	Ave Time	Ave red. %	MP Time	MP Iter	SP Iter
Ud1	x	53		17.2	5.6	6.2	x	73		10.5	10.3	10.3
	1	49	8.1	14.3	5.2	5.7	1	62	15.0	8.4	8.5	8.5
	2	53	-9.2	8.6	2.9	7.7	2	48	22.1	2.4	4.2	8.4
Ud2	x	187		109.3	6.1	6.8	x	111		16.6	8.4	8.4
	1	113	39.9	38.0	5.4	6.1	1	91	18.1	13.3	6.4	6.4
	2	91	19.5	25.0	2.4	5.8	2	88	3.5	6.3	3.7	7.4
Ue1	x	1500		1163.9	7.6	8.2	x	733		128.5	13.2	13.2
	1	751	50.0	436.4	6.6	7.4	1	575	21.5	92.6	10.0	10.0
	2	505	32.7	257.6	2.8	7.2	2	359	37.6	56.4	4.1	8.1
Ue2	x	2444		1911.0	5.7	6.7	x	1271		319.7	11.9	11.9
	1	1214	50.3	710.7	5.2	5.8	1	1046	17.7	358.7	7.5	7.5
	2	907	25.3	461.2	2.2	6.0	2	507	51.6	19.0	3.5	7.0
Uf1	x	11695[1]		10753.1	5.8	6.6	x	6227		3783.9	14.2	14.2
	1	10111[2]	13.5	9193.6	5.4	6.2	1	2181	65.0	946.9	7.8	7.8
	2	4717	53.3	3881.8	2.6	6.0	2	1541	29.3	380.5	4.0	8.0
Uf2	x	12634[3]		11104.8	4.5	5.5	x	3650		785.0	9.5	9.5
	1	10988[4]	13.0	9132.4	4.5	5.5	1	2920	20.0	793.8	6.5	6.5
	2	4143[5]	62.3	2428.1	2.0	6.0	2	1963	32.8	94.7	3.3	6.6

[1] The average of 5 instances.
[2] The average of 9 instances.
[3] The average of 2 instances.
[4] The average of 6 instances.
[5] The average of 7 instances.

In all cases, the improved UB helps reduce the total runtimes and the number of iterations. Runtime reduction ranges from 8.1-50.3% for the base BD algorithm and ranges from 15-65% for the ε-optimal BD algorithm. Even more pronounced performance improvement can be achieved when the good feasible solutions from the heuristic are not only used to improve UB, but also to generate Benders cuts. Up to 62.3% and 51.6% of additional runtime reductions are realizable through this practice. Although a negative percentage is reported for the class Ud1, we note that the additional Benders cuts are still successful in reducing the MP time and MP iterations. Since the improved UB alone already provides promising results for

these relatively small instances (compare to Ue1-2 and Uf1-2), the additional MP time saving is not significant enough to compromise the amount of time required to generate the additional Benders cuts. Thus, this leads to increased runtimes.

Based on results of the base BD algorithm, some instances in classes Uf1 and Uf2 are not solved within the 20000 second time limit. The majority of the huge runtimes contribute to solving the MP to optimality, which can be alleviated to a certain degree using the additional Benders cuts from the heuristic. However, the ε-optimal BD algorithm handles these larger instances more efficiently and, in most cases, more than 50% of runtimes can be reduced. Clearly, this significant savings is achieved as the MP is not optimized. Another important observation is that the heuristic can efficiently find improved solutions. Such improvement can be detected by comparing the last two columns in Table 23. The number of subproblem iterations being twice the number of the master problem iterations indicates that in almost all iterations, the heuristic can supply improved solutions to the subproblem.

In the third experiment, we examine the performance of the ε-optimal BD algorithm when applied to classes Uf1 and Uf2 under different $\Delta_1 - \Delta_2$ combinations. The algorithm can efficiently solve all instances to 2% optimality within 40 minutes as dictated in Table 24. It can be observed that the algorithm takes longer when the value of Δ_1 increases and takes less time with increased Δ_2. With increased Δ_2 (compare 20–40 and 30–50 with 20–50 and 30–60), signals can reach further RPs in the RP-network, thus setting up more RP-RP links that would be beneficial in allowing signals to transmit along their shortest possible path. As a result, improved UB can be achieved, which leads to shorter runtimes. On the other hand, fewer RPs (interpreted from MP solutions) are required to cover all nonRP nodes as Δ_1 increases. This, in turn, reduces the number of RP-RP links that can be established, and hence, worsens the upper bound solution (both from the dual subproblem and

from the heuristic). The smaller LB from a more relaxed MP, along with the inferior UB, leads to longer runtimes as illustrated when comparing setting 20–50 with 30–50. Finally, we emphasize that the upper bound heuristic can find improved feasible solutions in all settings of Δ_1–Δ_2 as dictated by the number of master problem and subproblem iterations.

Table 24: ε-optimal BD algorithm under different Δ_1–Δ_2 settings

Class	Δ_1–Δ_2	Ave Time	Ave MP Time	Ave MP Iter	Ave SP Iter	Ave MP Time/iter.	#RP	#Link
Uf1	20–40	1541.2	380.5	4.0	8.0	107.3	20	50
	20–50	1144.8	59.6	3.5	7.0	15.9	19	63
	30–50	2177.4	859.5	4.1	8.2	213.3	14	43
	30–60	1320.0	206.8	3.5	7.0	86.9	14	51
Uf2	20–40	1963.1	94.7	3.3	6.6	28.7	25	82
	20–50	1858.5	247.8	2.8	5.6	110.0	24	100
	30–50	2242.1	92.9	3.2	6.4	26.6	19	77
	30–60	2009.3	84.9	2.9	5.8	28.1	19	89

In the fourth experiment, we compare the RP-network from Model 3 with the network obtained from the "network design problem with relays" (MNDR) presented in Cabral et al. (2007). MNDR considers locating both the RPs and links in such a way that 1) the network construction cost (corresponding to locating RPs and links) is minimized, and 2) for each commodity, there exists a path linking the origin and destination in which the distance between the origin and the first RP, the last RP and the destination, and any two RPs are within the preset upper bound Δ_2. Cabral et al. (2007) develop 4 heuristic algorithms for solving this MNDR model and compare the solutions with lower bounds obtained from the path based formulation with a column generation approach. However, in this experiment, we formulate the MNDR using an arc based formulation and solve the formulation using the BC approach.

The objective function (5.41) is the total network construction cost in which the first term represents the total link set-up cost and the second term represents total RP location cost. Constraints (5.42) ensure that the path connecting a commodity's origin and destination satisfies the distance constraints. Constraints (5.43) are the flow conservation constraints. Constraints (5.44) permit the flow only on the established links. Constraints (5.45)-(5.46) locate RPs on the intermediate locations along the path connecting origins and destinations. Constraints (5.47) are the binary requirement of $\mathbf{x}$ and $\mathbf{z}$ variables and state that $\mathbf{y}$ are non-negative real variables.

$$\text{Min} \quad \sum_{k}\sum_{l} F_{kl}\, z_{kl} + \sum_{k} F_k\, x_{kk} \tag{5.41}$$

subject to

$$d_{kl}\, y_{kl}^{ij} \leq \Delta_2 \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k,l \in \mathcal{N} \tag{5.42}$$

$$\sum_{l} y_{kl}^{ij} - \sum_{l} y_{lk}^{ij} = \begin{cases} 1, & k = i \\ -1, & k = j \\ 0, & \text{o.w.} \end{cases} \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k,l \in \mathcal{N} \tag{5.43}$$

$$y_{kl}^{ij} + y_{lk}^{ij} \leq z_{kl} \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k,l \in \mathcal{N} \tag{5.44}$$

$$\sum_{l} y_{kl}^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k \in \mathcal{N}, k \neq i,j \tag{5.45}$$

$$\sum_{l} y_{lk}^{ij} \leq x_{kk} \qquad \forall\, [i,j] \in \mathcal{Q}, \forall\, k \in \mathcal{N}, k \neq i,j \tag{5.46}$$

$$x_{kk}, z_{kl} \in \{0,1\}, \;\; y_{kl}^{ij} \geq 0 \qquad \forall\, i,j,k,l \in \mathcal{N} \tag{5.47}$$

In this experiment, we generate the instance classes Ua1, Ub1, and Uc1, each with 10 instances. The results from solving the formulation (5.41)-(5.46) to a 3% optimality gap using CPLEX9.0 and the results from Model 3 are provided in Table 25; the values of Δ_1 and Δ_2 are fixed at 40. We note that MNDR allows the connections

between origins and destinations without locating RPs on the intermediate locations if the distances between them are shorter than Δ_2. Although formulation (5.41)-(5.46) does not consider the opportunity of making direct connections, we indirectly handle this issue by removing the commodities with distances between origins and destinations shorter than or equal to Δ_2 from the problem instances. Hence, all leftover commodities now require the intermediate RP locations, and the solutions of the MNDR and the formulation (5.41)-(5.46) are now the same. Based on this observation, we refer to the formulation (5.41)-(5.46) as MNDR.

Table 25: Comparison between different models

Class ($\|\mathcal{N}\|$-$\|\mathcal{Q}\|$)	Model	Total Cost	Cost^{xz}	Cost^{y}	#RP	#Link	$\bar{c}^M$	$\bar{c}^A$
Ua1 (20-67)	MNDR	174805	37500	137305	5.6	19.0	775	232
	Model 3	144670	39800	104870	5.9	6.5	572	320
Ub1 (25-105)	MNDR	283708	42000	241708	6.0	24.0	1216	329
	Model 3	217184	48250	168934	7.1	7.6	794	471
Uc1 (30-154)	MNDR	478804	48000	430804	6.7	29.0	1924	485
	Model 3	309220	61000	248220	9.2	13.0	851	430
Class ($\|\mathcal{N}\|$-$\|\mathcal{Q}\|$)	Model	d^A	$\bar{\Omega}^M$	$\bar{\Omega}^A$	$\bar{d}^M$	$\bar{d}^A$	L^M	L^A
Ua1 (20-67)	MNDR	86.1	2.5	0.67	206.3	137.70	6.9	4.43
	Model 3		1.2	0.25	166.1	104.95	5.6	3.54
Ub1 (25-105)	MNDR	88.3	3.3	0.87	262.6	155.21	8.9	5.07
	Model 3		1.4	0.25	174.1	107.76	5.6	3.74
Uc1 (30-154)	MNDR	90.2	4.9	1.24	322.5	186.67	11.0	6.10
	Model 3		1.4	0.21	175.6	107.62	6.1	3.90

In Table 25, Total Cost represents the total cost of implementing the MNDR and Model 3, which is the summation of the network construction cost (Cost^{xz}) and the total transmission cost (Cost^{y}). For the MNDR model, Cost^{xz} is the value of the objective function (5.41) and Cost^{y} is calculated from the value of $\mathbf{y}$ variables. For

Model 3, the total transmission cost is also obtained from the value of $\mathbf{y}$. However, for a fair comparison, the network construction cost of Model 3 now includes the link set-up cost for the connection between origins/destinations and RPs (originally, Model 3 only considers the fixed cost between RP-RP links). This cost is calculated from the value of $\mathbf{x}$ variables, which is equal to $\sum_i \sum_k F_{ik}\, x_{ik}\, i, k \in \mathcal{N}, i \neq k$. Moreover, the number of RPs and Links are represented by #RP and #Link, respectively.

We observe that the total cost of Model 3 is significantly less than that of the MNDR model. Recall that, MNDR only considers minimizing network construction costs. This, in turn, leads to a tree-shape (#Link = $|\mathcal{N}|$-1) RP-network with minimal Cost^{xz}; thus, large distances between leaf nodes can be expected. On the other hand, Model 3 considers both the network construction and the total transmission cost. Therefore the resulting RP-networks have more connectivity (more RPs and RP-RP links), which allows commodities to transmit on more direct, less circuitous, paths.

By partially having the total transmission cost in the objective function, the total transmission distances are also partially minimized; a low level of maximum and average transmission distances ($\bar{d}^M$ and $\bar{d}^A$), and maximum and average percentage circuitry levels ($\bar{\Omega}^M$ and $\bar{\Omega}^A$) can be expected. The additional connectivity also allows a more even utilization of the established links, as indicated by the lower values of maximum and average capacity usages, $\bar{c}^M$ and $\bar{c}^A$. Furthermore, transmission delays can occur while signals are passing through RPs (Cabral et al., 2008). In Table 25, L^M and L^A are the maximum and average numbers of transmission legs (number of times that signals are regenerated). Therefore, the maximum and average numbers of intermediate RPs is equal to L^M-1 and L^A-1, respectively. Thus, the transmission paths from Model 3 can potentially provide a better quality signal with less delay as there is a smaller number of RPs in the intermediate locations than in the MNDR.

Based on these observations, we conclude that Model 3 provides better RP-

networks that are not only cheaper but also facilitate better performance in terms of quality (shorter origin-destination distances and less intermediate RPs) and reliability (more connectivity).

V.1.5. Concluding Remarks

Model 3 considers the situation when a physical link must be established at a fixed charge prior to permitting the connection between any two relay points (common in telecommunications networks). In comparison to the other model in the literature, Model 3 provides better RP-networks that can facilitate better performance both in terms of quality and connectivity.

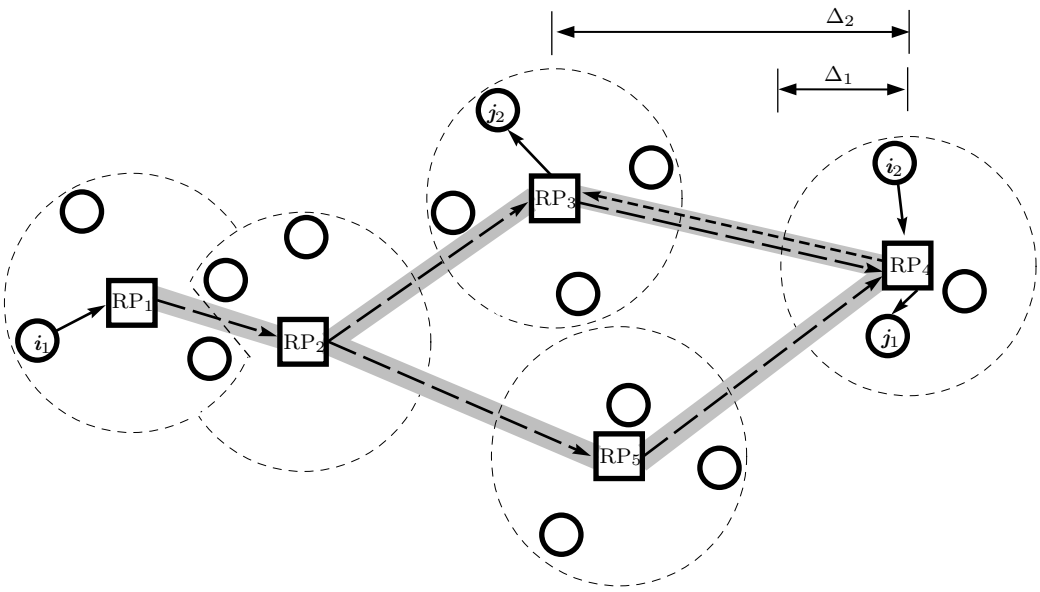
Exploiting the MIP formulation of Model 3, we observe that for a given network structure (relay locations, node assignments, and links established), the subproblem is a decomposable linear program. Similar characteristics are observed in Model 1 for which a variety of algorithms based on a Benders decomposition framework have shown promising performance. Thus, the same framework is also applied to the development of solution algorithms for Model 3.

While the cut disaggregation schemes and the ε-optimal framework can be directly applied to Model 3, the strong Benders cuts and the upper bound heuristic require refinements to address the different formulation of Model 3. In addition to the acceleration techniques applied to Model 1, we present a set of surrogate constraints that can reduce both the runtimes and number of iterations in a Benders framework. Moreover, we also enhance the algorithms using Benders cuts that are derived from improved heuristic solutions to handle the infeasibility from the master problem solution (in addition to extreme rays). The beneficial impacts of the accelerating techniques are illustrated in our extensive experimentations. Thus, the improved performance allows us to study the impact of modeling parameters on large

instances.

V.2. Model 4: RNDP with Fixed Link Set-up Cost and Capacity Constraints

Figure 8: A Schematic View of Model 4



Unlimited link capacity is an important assumption of Model 3. This assumption allows the signal to always transmit on its shortest path (sequence of RP-RP links) between the RPs to which the origin and destination are assigned. Thus, the RP-network under this assumption normally contains only one or a few alternative paths for each commodity, and many of them are shared by multiple commodities. In fact, too much traffic or signal flow in some areas (e.g., on links, between RPs, or along paths) can cause network congestion. This, in turn, induces transmission delay and, under extreme circumstances, can potentially lead to disconnection or network

failure (Vacca, 2001). Therefore, to avoid this situation and construct an effective RP-network, we include the capacity limitation on the established RP-RP links in order to control the total flow amounts and avoid traffic congestion. Hence, the resulting RP-network should have high connectivity and contain adequate transmission channels for commodities to share. With the inclusion of link capacity, Model 4 is a generalization of Model 3 and Model 4 reduces to Model 3 if the capacity limitation requirement on RP-RP links is removed.

The difference between Model 3 and Model 4 can be illustrated using Figure 8. The capacity of the RP-RP link (RP_3,RP_4) is shared by two commodities $[i_1, j_1]$ and $[i_2, j_2]$. If the capacity of this (RP_3,RP_4) link is not enough for both commodities, then fractions of one commodity must transmit through other channels. In this example, signals of commodity $[i_1, j_1]$ are transmitted using two paths, RP_1-RP_2-RP_3-RP_4 and RP_1-RP_2-RP_5-RP_4, because of the exhausted link (RP_3,RP_4).

V.2.1. The Model

To include the capacity limitation on the established RP-RP links, we incorporate the following constraints (5.48) into Model 3 and remove constraints (5.9) and (5.10), which are now redundant.

$$\sum_i \sum_j w_{ij} \left(y_{kl}^{ij} + y_{lk}^{ij}\right) \leq c_{kl} z_{kl} \qquad \forall\, k, l \in \mathcal{N},\ k < l \tag{5.48}$$

Note that a capacity limitation is also considered in Model 2 (Section IV.2.1), where it is stated using constraints (4.67), (4.72), and (4.73). Although both of them have identical effects, only $\frac{\mathcal{N}^2-\mathcal{N}}{2}$ constraints are required for constraints (5.48) while $(2\mathcal{Q}+1)(\frac{\mathcal{N}^2-\mathcal{N}}{2})$ are required for constraints (4.67), (4.72), and (4.73). Thus, using constraints (5.48) facilitates a better control of problem size because of the significantly smaller number of constraints. However, using constraints (5.48) or con-

straints (4.67), (4.72), and (4.73) would not affect the development of our Lagrangean decomposition algorithms as, after **z** variables are removed, constraints (5.48) would be reduced to constraints (4.67) and both alternatives still require constraints (4.72) and (4.73).

Incorporating constraints (5.48) into Model 3 and following the adjustment discussed above, the complete formulation of Model 4 is as follows:

$$\text{Min } Z = \sum_i \sum_k T_1\, d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij}\, y_{kl}^{ij} + \sum_k F_k\, x_{kk} + \sum_k \sum_l F_{kl}\, z_{kl} \tag{5.49}$$

subject to

$$d_{ik}\, x_{ik} \leq \Delta_1 \qquad \forall\, i,k \in \mathcal{N} \tag{5.50}$$

$$d_{kl}\, z_{kl} \leq \Delta_2 \qquad \forall\, k,l \in \mathcal{N} \tag{5.51}$$

$$\sum_m y_{mk}^{ij} - \sum_m y_{km}^{ij} = x_{jk} - x_{ik} \qquad \forall\, [i,j] \in \mathcal{Q},\, \forall\, k \in \mathcal{N} \tag{5.52}$$

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{5.53}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,k \in \mathcal{N} \tag{5.54}$$

$$z_{kl} \leq x_{kk} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{5.55}$$

$$z_{kl} \leq x_{ll} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{5.56}$$

$$\sum_i \sum_j w_{ij}\, (y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl} z_{kl} \qquad \forall\, k,l \in \mathcal{N},\, k < l \tag{5.57}$$

$$x_{ik},\, z_{kl} \in \{0,1\},\, y_{kl}^{ij} \geq 0 \qquad \forall\, i,j,k,l \in \mathcal{N} \tag{5.58}$$

As it also occurs in Model 2, the existence of link capacity destructs the decomposable structure of the **y** subproblem. Thus, applying Benders decomposition would require

the indecomposable subproblem to be solved in an aggregated form, and would be inefficient. Upon observing this characteristic of Model 4, we employ a different type of decomposition framework, Lagrangean relaxation, in the development of solution algorithms for Model 4. In our preliminary experiments, we also applied the Lagrangean decomposition framework (with the same copy constraints as in Model 2) to Model 4, however, we observe that the Lagragean relaxation algorithms provide better performance.

V.2.2. Lagrangean Relaxation Framework

Lagrangean relaxation (LR) has been extensively applied to complex MIP problems, especially in the context of uncapacitated (Holmberg and Hellstrand, 1998) and capacitated (Gendron and Crainic, 1994; Holmberg and Yuan, 2000) multicommodity network design problems (MND) (note that if $F_{kk} = 0, \forall k \in \mathcal{N}$, Model 4 is very similar to the capacitated MND). The basic framework of the LR approach involves relaxing a set of complicated constraints by 1) removing them from the constraint set, 2) multiplying the constraints with Lagrangean multipliers, and 3) incorporating the product of the constraints and the multipliers to the objective function. The attached term, corresponding to the relaxed constraints, to the objective function behaves as a penalty that arises whenever the relaxed constraints are violated. With part of the constraint set being removed, optimizing the relaxed formulation provides a lower bound to the original formulation.

The Lagrangean relaxation and upper bound procedures for Model 4 follow the same framework for solving the capacitated MND presented in Holmberg and Yuan (2000). However, the overall algorithms are different in many ways. Holmberg and Yuan (2000) incorporate the Lagrangean relaxation and upper bound procedures in a Branch-and-bound (BB) algorithm where the link variables ($\mathbf{z}$) are fixed to 0 or

1 in each BB node. Each time the BB algorithm attempts to construct the upper bound, the upper bound procedure is utilized only once, whether or not a feasible solution is found. In our algorithm, we emphasize more on trying to quickly find good feasible solutions. Instead of fixing the link variables, we develop a heuristic algorithm to adjust the link configurations whenever the upper bound procedure fails to find a feasible solution. With multiple settings of RP-network being input to the upper bound procedure, good feasible solutions can be expected even in the very early stages of our algorithm. Additionally, we incorporate various acceleration techniques to enhance the performance of our Lagrangean relaxation algorithm, especially when solving large problems. All these distinctions make our algorithms significantly different from the algorithm provided in Holmberg and Yuan (2000).

In the following sections, we provide a detailed discussion on the development of our LR algorithms based on the framework of Lagrangean relaxation.

V.2.2.1. Relaxed Formulation

In order to reduce the formulation size, we apply the preprocessing steps to the formulation (5.49)-(5.58). To do so, constraints (5.50)-(5.51) are now removed after setting $x_{ik} = 0$ if $d_{ik} > \Delta_1, \forall i, k \in \mathcal{N}$ and $z_{kl} = 0$ if $d_{kl} > \Delta_2, \forall k, l \in \mathcal{N}$. Additionally, we set $y_{kl}^{ij} = 0$ if $d_{kl} > \Delta_2, \forall [i, j] \in \mathcal{Q},\ \forall k, l \in \mathcal{N}$.

Exploiting the preprocessed formulation, we observe that multiple relaxation alternatives are applicable to Model 4. However, due to the large formulation size and the interrelationships between $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$, we are interested in relaxing only one constraint set that would allow further problem decomposition. Relaxing multiple sets of constraints is another alternative that would, very likely, lead to decomposable and easy-to-solve resulting formulation. Unfortunately, their benefits are usually compromised with the inferior lower bound strength as a large number of constraints

are removed. Therefore, in order to comply with our objective, the flow conservation constraints (5.52) is relaxed.

By letting λ_k^{ij} be the Lagrangean multiplier associated with constraints (5.52), the objective function is as follows:

$$\text{Min } Z = \sum_i \sum_k T_1 d_{ik} \sum_j \big(w_{ij} + w_{ji}\big) x_{ik} + \sum_i \sum_j \sum_k \sum_l T_2\, d_{kl}\, w_{ij} y_{kl}^{ij}$$

$$+ \sum_k F_k x_{kk} + \sum_k \sum_l F_{kl} z_{kl} + \sum_i \sum_j \sum_k \lambda_k^{ij} \big(x_{jk} - x_{ik} + \sum_l y_{kl}^{ij} + \sum_l y_{lk}^{ij}\big) \quad (5.59)$$

which is equivalent to

$$\text{Min } Z = \sum_i \sum_k \Big(T_1 d_{ik} \sum_j (w_{ij} + w_{ji}) + \sum_j (\lambda_k^{ji} - \lambda_k^{ij})\Big) x_{ik} + \sum_k F_k x_{kk}$$

$$+ \sum_k \sum_l F_{kl} z_{kl} + \sum_i \sum_j \sum_k \sum_l \Big(T_2\, d_{kl}\, w_{ij} + \lambda_k^{ij} - \lambda_l^{ij}\Big) y_{kl}^{ij} \quad (5.60)$$

We let A_{ik} and B_{kl}^{ij} represent the terms $\Big(T_1 d_{ik} \sum_j (w_{ij} + w_{ji}) + \sum_j (\lambda_k^{ji} - \lambda_k^{ij})\Big)$ and $\Big(T_2\, d_{kl}\, w_{ij} + \lambda_k^{ij} - \lambda_l^{ij}\Big)$, respectively. As a result, the relaxed formulation of Model 4 can be stated as follows:

$$\text{Min } Z = \sum_i \sum_k A_{ik}\, x_{ik} + \sum_k F_k\, x_{kk} + \sum_k \sum_l F_{kl}\, z_{kl} + \sum_i \sum_j \sum_k \sum_l B_{kl}^{ij}\, y_{kl}^{ij} \quad (5.61)$$

subject to

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{5.62}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i,k \in \mathcal{N} \tag{5.63}$$

$$z_{kl} \leq x_{kk} \qquad \forall\, k,l \in \mathcal{N} \tag{5.64}$$

$$z_{kl} \leq x_{ll} \qquad \forall\, k,l \in \mathcal{N} \tag{5.65}$$

$$\sum_i \sum_j w_{ij}\,(y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl} z_{kl} \qquad \forall\, k,l \in \mathcal{N} \tag{5.66}$$

$$x_{ik},\ z_{kl} \in \{0,1\},\ 0 \leq y_{kl}^{ij} \leq 1 \qquad \forall\, i,j,k,l \in \mathcal{N} \tag{5.67}$$

The relaxation of constraints (5.52) can break the tie between $\mathbf{x}$ and $\mathbf{y}$, and leave $\mathbf{y}$ variables to depend solely on $\mathbf{z}$ variables. This allows us to avoid solving for the $\mathbf{y}$ variable in an aggregated form, as presented in the next section.

V.2.2.2. Solving the Relaxed Problems

For a fixed value of $z_{kl}, k,l \in \mathcal{N}, k<l$, the optimal $y_{kl}^{ij}, [i,j] \in \mathcal{Q}$ can be obtained by solving the following subproblem:

$$\text{Min } E_{kl} = \sum_i \sum_j B_{kl}^{ij}\, y_{kl}^{ij} \tag{5.68}$$

subject to

$$\sum_i \sum_j w_{ij}\,(y_{kl}^{ij} + y_{lk}^{ij}) \leq c_{kl}\, z_{kl} \tag{5.69}$$

$$0 \leq y_{kl}^{ij} \leq 1 \qquad \forall\, i,j \in \mathcal{N} \tag{5.70}$$

If z_{kl} is 0, the RHS of constraint (5.69) becomes 0, and all y_{kl}^{ij} and y_{lk}^{ij}, $[i,j] \in \mathcal{Q}$ take the value of 0. On the other hand, if z_{kl} is 1, then the problem (5.68)-(5.70) is

essentially a 0-1 continuous knapsack problem, which can be solved efficiently using Algorithm 12.

Algorithm 12 Solving the subproblem (a 0-1 continuous knapsack problem)

1: For each (k,l), $k<l$, set $E_{kl}=0$, $C_{kl}=c_{kl}$, $y^{ij}_{kl}=y^{ij}_{lk}=0, \forall\,[i,j]\in\mathcal{Q}$;
2: **while** $C_{kl}>0$ **do**
3: $\quad\hat{B}^{ij}_{kl}=\min\{B^{ij}_{kl},B^{ij}_{lk}\}$;
4: $\quad$**if** $\hat{B}^{ij}_{kl}>0$ **then**
5: $\quad\quad$stop;
6: $\quad$**else**
7: $\quad\quad$Let $(\hat{i},\hat{j},\hat{k},\hat{l})$ be the indices associate with $\hat{B}^{ij}_{kl}$;
8: $\quad\quad E_{\hat{k}\hat{l}}=E_{\hat{k}\hat{l}}+\min\{\hat{B}^{ij}_{kl},(\hat{B}^{ij}_{kl}*\frac{C_{kl}}{w_{\hat{i}\hat{j}}})\}$;
9: $\quad\quad C_{kl}=C_{kl}-\min\{C_{kl},w_{\hat{i}\hat{j}}\}$;
10: $\quad\quad y^{\hat{i}\hat{j}}_{\hat{k}\hat{l}}=y^{\hat{i}\hat{j}}_{\hat{k}\hat{l}}+\min\{1,\frac{C_{kl}}{w_{\hat{i}\hat{j}}}\}$
11: $\quad\quad$Remove $\hat{B}^{ij}_{kl}$ from the consideration;
12: $\quad$**end if**
13: **end while**

In Algorithm 12, E_{kl} is the objective function of problem (5.68)-(5.70) defined on the link (k,l), $k,l\in\mathcal{N},k<l$. Considering the leftover link capacity of $C_{kl}=c_{kl}$, the algorithm tries to fill the knapsack (link capacity) with a commodity $[\hat{i},\hat{j}]$ in directions $(\hat{k},\hat{l})$ or $(\hat{l},\hat{k})$ that has the largest negative coefficient B^{ij}_{kl} (step 3). After determining the directions and the commodities for the knapsack, the objective function value E_{kl}, the leftover capacity C_{kl}, and the associated $\mathbf{y}$ variable are adjusted according to the flow amount which is the greater of the demand amount $w_{\hat{i}\hat{j}}$ or the leftover capacity C_{kl} (steps 7-10). Afterwards, the algorithm continues to find the next best commodity until the leftover capacity is exhausted and terminates (step 2). The algorithm also terminates when failing to find a commodity with negative coefficient B^{ij}_{kl} (step 4). We note that, apart from the difference in B^{ij}_{kl} expression, Algorithm 12 is the same as Algorithm 4 (Section IV.2.2.3).

After $\mathbf{y}$ is determined, the associated cost E_{kl} can be incorporated into the objective function of the relaxed problem to represent the contribution of $\mathbf{y}$ in the total cost. In doing this, constraints (5.66) can be removed from the formulation and the relaxed problem can now be re-stated as follows:

$$\text{Min } Z_{LBP} = \sum_i \sum_k A_{ik}\, x_{ik} + \sum_k F_k\, x_{kk} + \sum_k \sum_l (F_{kl} + E_{kl})\, z_{kl} \tag{5.71}$$

subject to

$$\sum_k x_{ik} = 1 \qquad \forall\, i \in \mathcal{N} \tag{5.72}$$

$$x_{ik} \leq x_{kk} \qquad \forall\, i, k \in \mathcal{N} \tag{5.73}$$

$$z_{kl} \leq x_{kk} \qquad \forall\, k, l \in \mathcal{N} \tag{5.74}$$

$$z_{kl} \leq x_{ll} \qquad \forall\, k, l \in \mathcal{N} \tag{5.75}$$

$$x_{ik},\ z_{kl}\ \in \{0, 1\} \qquad \forall\, i, k, l \in \mathcal{N} \tag{5.76}$$

Clearly, the problem (5.71)-(5.76) is significantly smaller than the original problem (5.49)-(5.58), as the portion of the formulation corresponding to $\mathbf{y}$ is embedded in the coefficient of $\mathbf{z}$. Solving this formulation provides a valid lower bound to the original problem; hence, it will be referred to hereafter as the "lower bound problem" or "LBP." Moreover, due to its reduced size, the Branch-and-cut approach (CPLEX 9.1) can now effectively solve the LBP.

V.2.2.3. Accelerating the Relaxed Problems

Although LBP has been reduced to a manageable size that is solvable by the Branch-and-cut approach, the runtime is still growing at a considerably fast rate. In this case, the surrogate constraints (5.38) introduced in Section V.1.3.2 can help speed up LBP. Constraints (5.39) and (5.40) were also tested in our preliminary experiment; however,

their algorithmic enhancements were not observed.

In addition to the surrogate constraints, we also consider applying the "early stopping criteria" to accelerate the solution time of LBP. Detailed discussion of this technique in Benders decomposition can be found in Üster et al. (2007). Without completing the optimization process, this technique involves solving the problem to a small optimality gap, thus a range that contains the optimal solution can be acquired. Specifically, for Model 4, LBP is stopped whenever the optimality gap reaches 1.5%. Then, a lower bound of the lower bound problem ($\underline{Z_{LB}}$) can be achieved from the Branch-and-cut approach and used in the rest of our algorithm in place of Z_{LB}. The purpose of solving LBP is to obtain the lower bound of Model 4, therefore $\underline{Z_{LB}}$, the lower bound of LBP, is also a valid lower bound of Z. Moreover, the partial optimization can help avoid the tail-off effect; it takes a considerable amount of time to close the final few percentage points of the optimality gap.

V.2.3. Upper Bound Heuristic

With the flow conservation constraints (5.52) being relaxed, the solution from LBP is not generally composed of $\mathbf{y}$ variables that define valid transmission paths. Therefore, the commodities must be re-transmitted during the re-construction of the RP-induced network given from LBP. Note that the configuration of the RP-network is implied by the value of $\mathbf{x}$ (RP locations and node assignments) and $\mathbf{z}$ (RP-RP links established) variables.

Due to the similar structural characteristics between Model 3 and Model 4, we devise the upper bound heuristic (UBH), Algorithm 11 in Section V.1.3.3, to re-construct the RP-network, thus generating feasible solutions from LBP solutions and obtaining valid upper bounds. However, the upper bound procedures $UB(\cdot)$ between Model 3 and 4 are not the same due to the link capacity. In this section, we

will discuss the development of Algorithm 13, the upper bound procedure $UB(\cdot)$, for utilization as part of the upper bound heuristic UBH.

Algorithm 13 Upper bound procedure for Model 4: $UB(\cdot)$

Set $Z_{UB} = 0$, $U_{kl} = 0$ and $C_{kl} = c_{kl}$, $k, l \in \mathcal{N}, k < l$;
Sort all commodities in descending order of $\lambda^{ij}_{r(j)} - \lambda^{ij}_{r(i)}, \forall [i,j] \in \mathcal{Q}$;
for each sorted commodity $[i,j]$ **do**
  Let $W_{ij} = w_{ij}$
  while $W_{ij} > 0$ **do**
    Solve the shortest path from $r(i)$ to $r(j)$ using the distance matrix D;
    Let $\bar{\mathcal{A}}$ be the set of arcs in the shortest path and $L_{ij} = \sum_k \sum_l D_{kl}$, $(k,l) \in \bar{\mathcal{A}}$;
    $f_{ij} = \min\{W_{ij}, \min\{C_{kl} : (k,l) \in \bar{\mathcal{A}}, D_{kl} < M\}\}$;
    $Z_{UB} = Z_{UB} + (T_2 \times L_{ij} \times f_{ij})$;
    $W_{ij} = W_{ij} - f_{ij}$;
    for $(k,l) \in \bar{\mathcal{A}}$ **do**
      if $k < l$ and $D_{kl} < M$ **then**
        $C_{kl} = C_{kl} - f_{ij}$;
      end if
      if $k > l$ and $D_{kl} < M$ **then**
        $C_{lk} = C_{lk} - f_{ij}$;
      end if
      if $C_{kl} = 0$ **then**
        Set $D_{kl} = D_{lk} = M$;
      end if
      $U_{kl} = U_{kl} + 1$;
    end for
  end while
end for
$Z_{UB} = Z_{UB} + \sum_i \sum_k T_1 d_{ik} \sum_j \left(w_{ij} + w_{ji}\right) x_{ik} + \sum_k F_k x_{kk} + \sum_k \sum_l (F_{kl} \times \min\{z_{kl}, U_{kl}\})$;

In order to construct an upper bound, Algorithm 13 selects one commodity at a time and solves for the shortest path to send flows. However, the order in which the commodities are selected can affect the upper bound quality. In Model 3, such an order is not a significant issue due to unlimited link capacity. However, this is not the case in Model 4, and we observe that good upper bound quality can be achieved if

the commodities are sorted in descending order of $\lambda^{ij}_{r(j)} - \lambda^{ij}_{r(i)}$, using the most recent $\boldsymbol{\lambda}$ value (step 2). Recall that $r(k)$ are the RPs to which node k is assigned, and D represents the distance matrix in which the entry D_{kl} equal to d_{kl} if $Z_{kl} = 1$ and M if $Z_{kl} = 0$. For each sorted commodity $[i, j]$, the shortest path from $r(i)$ to $r(j)$ is calculated using the distance matrix D. The length of the shortest path is denoted by L_{ij} and the set of arcs in the shortest path is denoted by $\bar{\mathcal{A}}$. The amount of flows f_{ij} sent along this shortest path is equal to the lesser of the leftover capacity W_{ij} or the minimum leftover link capacity along the shortest path (excluding the artificial links and the exhausted links). f_{ij} and L_{ij} are then used to adjust the total cost Z_{UB} (step 9), the leftover demand W_{ij} (step 10), and the leftover capacity C_{kl} (steps 12-20). Whenever the capacity of a link is exhausted, the associated distance D_{kl} in D matrix is set to M (step 19). Moreover, the link utilization matrix U is also maintained (step 21) as it is devised in the UBH (see Section V.1.3.3). The procedure continues until all the commodities are re-transmitted and it terminates. Finally, the cost of the RP locations, links set-up (only if used), and transmission between nonRP nodes and RPs are included in Z_{UB} (step 25).

V.2.4. Subgradient Method

Lagrangean multipliers play an important role in our LR algorithm. Note that, if the optimal λ is given, then the values of $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ obtained from LBP would also be the optimal solution. However, since their optimal values are not known, we search for a good candidate of the Lagrangean multipliers iteratively. In the following discussion, we describe the subgradient optimization (see for example Fisher, 1981) used in Model 4 to update λ based on the solutions of LBP and UBH.

At the end of each iteration (after solving LBP and applying UBH), if the optimality gap between the best lower and upper bounds is not small enough, then

λ is updated as follows:

1. Let $\mathbf{x^t}$ and $\mathbf{y^t}$ be the solutions from LBP in iteration t and let the search direction v_k^{tij} be $\left(\bar{x}_{jk}^t - \bar{x}_{ik}^t + \sum_l \bar{y}_{kl}^{tij} + \sum_l \bar{y}_{lk}^{tij}\right)$.

2. Let Z_{LB}^t be the lower bound obtained in iteration t, LB_{best} and UB_{best} be the best lower and upper bounds found until iteration t. Then the step size $s^t = f^t \times \frac{UB_{best} - Z_{LB}^t}{\sum_i \sum_j \sum_k v_k^{tij\,2}}$ where f^t is the step size factor in iteration t,

3. Finally, we set $\lambda_k^{(t+1)ij} = \lambda_k^{tij} + (ss^t \times v_k^{tij})$.

Initially f^0 is set to 1.8, and is multiplied by 0.4 whenever UB_{best} is not updated for 40 consecutive iterations. Note that the above procedure is for updating λ at the end of each iteration. For the first iteration, one possible alternative is to set λ to $\mathbf{0}$. However, from our preliminary experiment, we found that the convergence of the LR algorithm can be improved by setting the initial λ as follows:

1. Assume that $x_{kk} = 1, \forall k \in \mathcal{N}$ and define a distance matrix D with entry $D_{kl} = d_{kl}$ if $d_{kl} \leq \Delta_2$ and $D_{kl} = M$ otherwise.

2. Solve the all pair shortest paths problem over the distance matrix D (by Dijkstra's algorithm) and let L_{kl} be the length of the shortest path between node k and node l.

3. Then, the initial Lagrangean multiplier is set to $\lambda_k^{0ij} = w_{ij} \times L_{ik}$.

We observe that setting the initial λ as discussed above not only provides a good starting optimality gap (through improved initial lower bound) but also helps reduce the number of LR iterations (via good starting λ).

V.2.4.1. Initial Upper Bound

Generally, UB_{best} is weak in the first (initially set to a large number) and early iterations (after only a few updates) of the LR algorithm. As UB_{best} is employed in the subgradient optimization, its unrealistically large value can be misleading and cause the subgradient to provide inferior λ. This leads to a poor performance of the overall algorithm. On the other hand, a good starting UB_{best} would not only help with the optimality gap, but also help the subgradient optimization in adjusting the the value of λ.

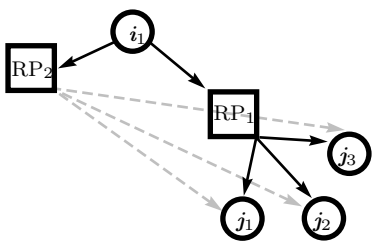
In order to obtain a good initial upper bound, we consider using the $UB(\cdot)$ from Section V.2.3 to construct an initial heuristic algorithm for the finding of good initial solutions. To do this, we first define O as a set of opened RPs and let S_O be the set of opened links corresponding to O. In S_O the link (k, l) is opened if $x_{kk} = x_{ll} = 1$ and $d_{kl} \leq \Delta_2$. Moreover, let d^e_{kl} be the Euclidean distance between nodes k and l. Based on these representations, the initial heuristic, Algorithm 14, is presented below.

Algorithm 14 Initial heuristic algorithm for Model 4

Set $O = \bar{O} = \mathcal{N}$;
Let $S_{\bar{O}}$ be the set of opened links associated with $\bar{O}$;
Calculate $UB(S_{\bar{O}})$ and set $UB_{best} = S_O$;
while $|O| > 0$ **do**
  for $o \in O$ **do**
    Find $S_{\bar{O}\setminus o}$ and reassign the nodes that are previously assigned to o;
    if All nonRP nodes can be reassigned **AND** $UB(S_{\bar{O}\setminus o}) < UB_{best}$ **then**
      $UB_{best} = UB(S_{\bar{O}\setminus o})$;
      $O = \bar{O} = \bar{O} \setminus o$;
    else
      $O = O \setminus o$;
    end if
  end for
end while

Algorithm 14 first assumes that all nodes imply RPs and it initializes the set of located RPs O and $\bar{O}$ (step 1). By utilizing $S_{\bar{O}}$, the set of opened links corresponds to $\bar{O}$, and UB_{best} is set equal to $UB(S_{\bar{O}})$ (step 3). In the search for an improved solution, the algorithm closes one of the opened RP o, and obtains a new set of opened links $S_{\bar{O}\setminus o}$. Whenever an RP is closed, the nodes previously assigned to the RP (including itself) must be reassigned to another RP within Δ_2 distance. In this case, we choose to reassign nonRP node i to RP k s.t. $k = \text{argmin}\Big\{ \sum_j \Big((w_{ij} + w_{ji}) \times (d_{ik} + d^e_{kj}) \Big) : x_{kk} = 1 \text{ and } d_{ik} \leq \gamma_1 \Big\}$. This type of assignment permits the majority of commodities to and from node i to transmit in the most direct fashion. We illustrate this assignment using Figure 9, where it is better if node i_1 is assigned to RP_1 rather than RP_2. If there is no RP within a Δ_1 reach from any nonRP node, then the new set of opened RPs is infeasible and the algorithm re-opens o and closes another RP. If the assignment process is complete, then the algorithm calculates $UB(S_{\bar{O}\setminus o})$ (step 7). If the new solution (set of opened RPs and established links) improves UB_{best}, then UB_{best} is replaced with $UB(S_{\bar{O}\setminus o})$ and the set of opened RPs are updated (steps 8-9). Otherwise, the RP o is removed from consideration (step 11) and another opened RP is closed. The algorithm continues in this fashion and terminates when all the opened RPs are tried and the set O is empty.

Figure 9: Assignment of Nodes to RPs



V.2.5. Overall Framework

The overall procedure of our LR algorithm is presented in Algorithm 15. We note that the LR algorithm in this section is based on the same framework as the LD algorithm in Section IV.2.4.

Algorithm 15 Lagrangean relaxation algorithm for Model 4: LR1

1: Set $LB_{best} = 0$, $UB_{best} = M$, $Iter = t_{ni} = 0$;
2: **while** $f^t > \varepsilon_f$ **do**
3: $Iter = Iter+1$; $t_{ni} = t_{ni} + 1$;
4: **Solve** LBP **for** Z^t_{LB};
5: **if** $Z^t_{LB} > LB_{best}$ **then**
6: $LB_{best} = Z^t_{LB}$
7: $t_{ni} = 0$
8: **end if**
9: **if** $t_{ni} = ni$ **then**
10: $f^t = f^t \times m^f$;
11: $t_{ni} = 0$
12: **end if**
13: **Apply the upper bound heuristic to obtain** Z^t_{UB};
14: **if** $Z^t_{UB} < UB_{best}$ **then**
15: $UB_{best} = Z^t_{UB}$
16: **end if**
17: **if** $(\frac{UB_{best} - LB_{best}}{UB_{best}} \leq \varepsilon_{opt})$ **then**
18: Stop;
19: **end if**
20: **Update the** λ **using the subgradient optimization**;
21: **end while**

The algorithm starts by setting LB_{best} to 0 and UB_{best} to a large number M (step 1). Then the algorithm solves LBP for Z^t_{LB} (step 4) and inputs $\mathbf{x}$ and $\mathbf{z}$ to UBH for the construction of Z^t_{UB} (step 13). If $Z^t_{LB} > LB_{best}$, then LB_{best} is updated with Z^t_{LB} (steps 5-6). Likewise, if $Z^t_{UB} < UB_{best}$, then UB_{best} is replaced by Z^t_{UB} (steps 14-15). Each time, after solving for the upper bound Z^t_{UB}, the optimality gap

$\left(\frac{UB_{best}-LB_{best}}{UB_{best}}\right)$ is calculated and the algorithm terminates if the gap is smaller than ε_{opt} (steps 17-18). Otherwise, if the gap still larger than ε_{opt}, then the subgradient optimization is devised to update the Lagrangean multipliers (step 20) and repeat the overall process. Note that the step size factor f^t, which is utilized in the subgradient optimization, is multiplied by m^f (0.4) if the LB_{best} is not improved for ni (40) consecutive iterations (steps 9-10). Hence, as f^t becomes very small, the differences between λ in two successive iterations are insignificant, thus leading the same lower bound solution being generated repeatedly. Therefore, the algorithm also terminates when $f^t \geq \varepsilon_f$ (step 2).

We refer to the algorithm just described as "LR1." When the initial heuristic (Section V.2.4.1) is incorporated in step 1 of Algorithm 15, the algorithm is then referred to as "LR2." According to the early discussion, LBP experiences a tail-off effect when solving large instances, especially with increased $|\mathcal{N}|$. Therefore, the surrogate constraints (5.38) and the early stopping criteria are utilized to help accelerate the solution time of LBP. Thus, "LR3" is used for representing LR2 with the surrogate constraints (5.38). Finally, if the early stopping criteria is applied to LR3, then the algorithm is denoted using "LR4."

V.2.6. Computational Experiments

In this section, we provide a detailed experiment to evaluate the algorithmic performance of the LR algorithm and to identify the beneficial impact of the accelerating techniques for Model 4. Unless stated otherwise, our algorithms (LR1-4) and the Branch-and-cut (BC) approach are assigned a preset time limit of 2 hours (7200 seconds).

In the first experiment, we benchmark our LR algorithm with the results obtained from solving problem classes Ua1, Ub1, Uc1, and Ud1 with the BC approach.

The same 10 instances from the first experiment in Section V.1.4 are solved with Δ_1–Δ_2 set to 20–40. For the link capacity c_{kl}, we set c_{kl}, $k, l \in \mathcal{N}$ – based on the implied link utilization level $\bar{c}_{kl}$ calculated from the values of $\mathbf{y}$ in Section V.1.4 – in such a way that the capacity constraints (5.57) are forced active. Specifically, c_{kl} is equal to 600, 700, 900, and 1100 for the classes Ua1, Ub1, Uc1, and Ud1, respectively. However, due to the inflexible capacity requirement and limited connectivity, especially in small instances, only 6, 8, and 9 instances remain feasible in the classes Ua1, Ub1, and Uc1 under the tight capacity setting.

Table 26: Comparing LR3 with BC approaches

				Ave Time			Ave Gap				
Class	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{Q}\rvert$	c_{kl}	$BC_{0\%}$	$BC_{2\%}$	LR3	LR3	$Gap^{lb}_{0\%}$	$Gap^{ub}_{0\%}$	$Gap^{lb}_{2\%}$	$Gap^{ub}_{2\%}$
Ua1	20	80	600	5	4	35	1.95[1]	1.73	0.23	1.41	0.23
Ub1	25	125	700	46	24	86	1.92[2]	1.49	0.44	1.19	0.31
Uc1	30	180	900	95	46	19	1.92	1.58	0.34	1.24	0.18
Ud1	40	320	1100	2043	749	178	1.94[1]	1.14	0.81	0.68	0.69

[1] One instance terminates as ε_f become very small.
[2] Two instances terminate as ε_f become very small.

In Table 26, columns 5 and 6 are the BC runtimes for solving the formulation of Model 4 (after constraints (5.50)-(5.51) are preprocessed) to optimality and to 2% optimal. We observe that the runtimes grow extremely fast with increasing $|\mathcal{N}|$, even when they are optimized to only 2% optimal. Although instances with more than 40 nodes (i.e., class Ue1 and larger) are not solvable with the BC approach, their runtimes are expected to be very large following the rapid growth we observed. On the other hand, the runtimes of the LR algorithm (LR3) grow at a much slower rate. Aiming at a 2% optimality gap (set ε_{opt} to 2%), LR3 takes less than 3 minutes on average to solve these instances (see column 7 for runtimes and column 8 for implied optimality gap). However, there are some instances that LR3 algorithm terminates as ε_f become very small prior to reaching 2% optimality.

Columns 9-10 and 11-12 measure the quality of the LB_{best} and UB_{best} with respect to the optimal and the 2% optimal solutions from the BC approach. Clearly, the heuristic algorithm in LR3 can provide near optimal solutions that are between 0.22-0.82% from optimality and only between 0.18-0.69% from the BC 2% upper bounds. In terms of lower bound strength, LB_{best} is slightly inferior to UB_{best} since its gaps are around 1.14-1.73% from optimality and 0.68-1.41% from the BC 2% lower bounds.

In the second experiment (see Table 27), we compare the performance of different LR algorithms, LR1-LR4, in solving instances of classes Ud1-2, Ue1-2, and Uf1-2. In each class, 10 instances (the same instances from Section V.1.4) are generated and solved with Δ_1–Δ_2 fixed at 20–40. With increasing number of nodes in larger instances, the underlying networks are more connective with respect to Δ_1 and Δ_2. As a result, instances are more flexible in terms of a capacity requirement, and are now solvable even under a tight capacity setting (capacity c_{kl} is reported in column 4).

Table 27: Comparing different LR algorithms

				LR1		LR2		LR3		LR4	
Class	$\|\mathcal{N}\|$	$\|\mathcal{Q}\|$	c_{kl}	Gap	Time	Gap	Time	Gap	Time	Gap	Time
Ud1	40	320	1100	11.93[5]	471	1.93[1]	187	1.94[1]	178	1.98[1]	115
Ud2	40	640	1100	12.21[5]	508	2.18[2]	447	2.19[1]	363	2.21[3]	382
Ue1	60	720	2000	7.12[5]	4169	1.90	899	1.93	310	1.93	353
Ue2	60	1440	2000	12.25[5]	1960	1.90	978	1.95	680	1.96	807
Uf1	80	1280	3000	6.55[5]	7738	3.18[5]	7635	2.00[2]	4471	1.95[2]	3700
Uf2	80	2560	3000	9.32[5]	7355	2.28[4]	6949	1.97[1]	3058	1.89	3487

[1] One instance terminates with optimality gap greater than 2%.
[2] Two instances terminate with optimality gap greater than 2%.
[3] Three instances terminate with optimality gap greater than 2%.
[4] Eight instances terminate with optimality gap greater than 2%.
[5] Every instance terminates with optimality gap greater than 2%.

In the early stages of the LR algorithm, the lower bound solutions generally imply infeasible RP-networks from which the upper bound heuristic cannot construct

good feasible solutions (and upper bounds). With the absence of realistic UB_{best}, the subgradient cannot justify the Lagrangean multipliers effectively, thus leading to poor algorithmic performance. In this case, providing good UB_{best} to the subgradient optimization is essential, as illustrated by comparing the results of LR1 with LR2. The gaps of LR1 range between 6.55–12.25% as opposed to between 1.90–3.18% for LR2. Clearly, the initial heuristic can significantly help reduce both the average gap and the associated runtimes.

The performance of LR2 is still inadequate as the optimality gap remains above 2% (3.18% for Uf1 and 2.28% for Uf2) and very large runtimes are reported. The majority of the runtimes are spent solving LBP, which, in most cases, experience tail-off effects. To this end, the surrogate constraints (5.38) become beneficial as illustrated by comparing columns 7 with 9. Although there is no significant difference in terms of gap for Ud1-2 and Ue1-2, the runtimes can be reduced by considerable amounts. In fact, the gap for large problem classes Uf1 and Uf2 are now below 2%, and almost half of the runtimes can be saved. Finally, comparing LR3 and LR4 illustrates the benefit of the early stopping criteria in further reducing the optimality gap for large instances, classes Uf1 and Uf2. Note that the runtimes of LR4 are higher than those of LR3 for 4 out of 6 problem classes, however, the runtime for problem class Uf1 (this class requires the longest runtime) can be reduced by about 20%.

Due to their algorithmic enhancement, we assume the use of the initial heuristic, surrogate constraints, and early stopping criteria in the next experimentation.

In the third experiment, we examine the performance of LR4 and the solution characteristics under different Δ_1–Δ_2 settings. The results are reported in Table 28. Note that T^{lb} and T^{ub} represent the total time spent on solving the LBP and UBH, #RP and #Link represent the number RPs and RP-RP links located by Model 4, respectively.

With c_{kl} fixed at 3000, LR4 is capable of solving most instances to below 2% optimality in all settings. For the instances that LR4 terminates before 2% optimality, the average gap is as low as 2.15% with the maximum gap being 2.33%. Although the runtimes range between 2600-6000 seconds, UBH takes only around 200-1000 seconds (with an average of 411 seconds) to provide very good feasible solutions (referencing our observation in the first experiment, the true quality of the upper bounds are usually much smaller than the optimality gap).

Table 28: LR4 under different Δ_1–Δ_2 settings

Class	$\Delta_1 - \Delta_2$	Gap	Time	T^{lb}	T^{ub}	#RP	#Link	$\bar{c}^A$	$\bar{c}^M$
Uf1	20–40	1.95[1]	3700	2998	332	20	54	898	2786
	20–50	1.96	3327	2729	216	19	63	629	2321
	30–50	2.09[2]	5466	4775	245	14	43	820	2447
	30–60	2.02[2]	5925	5058	272	13	46	678	2408
Uf2	20–40	1.89	3487	1903	913	28	107	954	3000
	20–50	1.93	2617	1585	429	25	111	750	2832
	30–50	1.96	2855	1759	425	19	82	927	3000
	30–60	1.96	3306	2054	453	19	93	734	2667

[1] Two instance terminates with optimality gap greater than 2%.
[2] Five instances terminate with optimality gap greater than 2%.

In terms of solution characteristics, fewer RPs are required to cover all the nodes in the service region when the value of Δ_1 increases. In consequence, fewer RP-RP links can be established, and the resulting RP-network now has increased link utilization levels $\bar{c}^A$ and $\bar{c}^M$. On the other hand, increasing Δ_2 leads to lowered $\bar{c}^A$ and $\bar{c}^M$. With increased Δ_2, signals can travel further in the RP-network and provide the RP-network with additional available links (even with fewer RP locations). Due to the increased connectivity, it is now beneficial to set up more RP-RP links and allow commodities to travel in their shortest possible routes.

V.2.7. Concluding Remarks

Model 4 further extends the base model to include the capacity limitation on an established link (in addition to the fixed link set-up cost). By assuming unlimited or very large link capacity, Model 3 is a special case of Model 4 where efficient algorithms based on Benders decomposition have already been developed. However, the existence of capacity destroys the decomposable structure of the subproblem, thus applying a Benders decomposition framework to Model 4 now appears ineffective. A similar situation is also observed in Model 2, where we apply the Lagrangean decomposition framework for the development of efficient solution algorithms.

We observe that applying Lagrangean relaxation to only one set of constraints (the flow conservation constraints) can facilitate the decomposition of the relaxed problem and, at the same time, maintain the majority portion of the structural requirements; hence, tight lower bounds can be obtained. In order to construct good feasible solutions, the upper bound procedure and heuristics are applied to the lower bound solutions. Coupled with the initial upper bound heuristic and one set of constraints from Model 3, we developed solution algorithms capable of systematically solving large instances of Model 4 under tight capacity and various parameter settings to small optimality gaps within reasonable runtimes.

CHAPTER VI

CONCLUSIONS AND FUTURE DIRECTIONS

Service industries make up a major component of the U.S. economy. Among them, the full truckload trucking and telecommunications industries have very important roles in their industry sectors.

In the full truckload trucking industry, most truck providers have suffered the problem of very high driver turnover that has continuously occurred for many decades. Numerous approaches have attempted to alleviate this turnover problem, but failed to provide long term solutions, as the cause of the problem is the very nature of the work itself. Under the typical dispatching method (PtP method), most truck drivers are assigned a long tour length journey that keeps them on the road for an extended period of time, which eventually leads drivers to quit their jobs. Having observed a very low turnover rate in the less-than-truckload (LTL) industry, we propose relay network and relaying operations that closely resemble the dispatching methods applied in LTL trucking, in order to improve drivers' job satisfaction and help truck providers retain their drivers. We expect a reduced driver turnover from utilizing the relay network as it would provide truck drivers with more regularized driving routines, similar to those of LTL drivers.

To this end, we develop two models that not only facilitate the reduction of tour length, but also take into account factors affecting performance such as, load-imbalance, link-imbalance, percentage-circuitry, and capacity limitation. Load-imbalance and link-imbalance serve the objective of controlling empty mileage, while percentage-circuitry provides the control of extra travel distance (from relaying shipments over the network). Finally, capacity limitation helps with the planning of workforce, resource (equipment), and traffic. We model the construction of relay networks with

these requirements using two mathematical formulations. Load-imbalance and percentage circuitry are included in Model 1, whereas link-imbalance and capacity limitation are included in Model 2. The formulations of both models are extremely large in size, hence solving them with the typical Branch-and-cut approach appears ineffective. Therefore, to obtain solutions for these two models, we develop solution algorithms based on Benders decomposition (BD) for Model 1, and Lagrangean decomposition (LD) for Model 2. To enhance the performance of the BD algorithm, we employ the strengthened Benders cuts, cut disaggregation schemes, ε-optimal framework, and heuristics algorithm. For the LD algorithm, we define the copy constraints in aggregate form for better control of formulation size and the decomposition of the relaxed problem. For both models, we also develop heuristics algorithms to potentially convert the lower bound solutions into good feasible solutions providing tight upper bounds. All techniques provide algorithms that can solve relatively large instances to small optimality gaps within a reasonable period of time. The efficacy of our algorithms is illustrated through extensive experimentation. From our experimental results, we also observe the impacts of problem parameters on both the algorithmic performance and solution characteristics.

On the other hand, the motivation to apply relay network to the telecommunications industry is from the physical limitation of signal. Because signal fades with distance, repeaters must be located over a large service region to amplify signal strength whenever it is transmitted beyond its transmission range. Additionally, other equipment may be required so as to reduce noise or to connect different frequency cables. The location of this equipment is based on proximity, which makes this type of transmission network coincide with our concept of relay network. To better capture the structural requirement of the telecommunications network, we incorporate into our Model 3 the links selection with fixed cost (cable installation) and into Model 4

the link capacity (limited bandwidth). Considering the location of relay points and links, the single assignment, and the routing decisions, Models 3 and 4 integrate the key characteristics of the uncapacitated single assignment hub location problem and the uncapacitated and capacitated multicommodity network design problem. The integrated products (Models 3 and 4) are very general models that are difficult to solve. Due to their similar uncapacitated network structures, the BD algorithms that show promising performance in solving Model 1 are applied to solve Model 3. To handle the infeasibility from the master problem, the BD algorithms are further enhanced by surrogate constraints and the Benders cut derived from the improved heuristics solutions. For Model 4, the capacitated version of Model 3, Lagrangean relaxation (LR) based algorithms are developed. The surrogate constraints and heuristics algorithms developed for Model 3 are also utilized in Model 4 in order to achieve algorithmic improvements. Again, the BD and LR algorithms for Models 3 and 4 are efficient and their performance, along with the beneficial impacts of each accelerating technique, are indicated in our computational studies.

In conclusion, we have developed, in total, four mathematical models for the design of different relay networks customized to meet the requirements for full truckload transportation and telecommunicatons applications. For each of the models, we have developed solution algorithms to reflect these distinct characteristics, thus improving their performance to make them capable of effectively solving large instances.

VI.1. Contributions

The significance of service industries has grown continuously, especially in the telecommunications and full truckload industries. Due to increased competition and growing demand, every service provider must improve its performance and efficiently con-

trol its cost effectiveness. For this purpose, we have studied the effective design of relay networks for service providers. The contributions of this dissertation can be summarized as follows.

1. The application of relay networks in the full truckload trucking industry aims to address an existing industry problem: high driver turnover. Utilizing a relay network is potentially a long term solution to this problem as it can provide a more regularized driving routine and increase the go-home rate for truck drivers. In fact, the operation of truck drivers will then be altered and more similar to those in the less than truckload industry, in which very low turnover rates are reported. However, the reduced turnover rate from implementing the relay network may be compromised by the empty mileage and extra travel distances, an additional burden to truck providers. Thus, our models also control them to low levels.[1] This research provides mathematical models and solution approaches for the design of a cost effective relay network that could achieve these objectives.

2. The applications for telecommunications and related industries provide more realistic mathematical models for designing transmission networks. We capture the physical limitations (e.g., restricted transmission range and capacity) and other general requirements (e.g., repeaters and cable installation) in long distance telecommunication using models that combine key characteristics of the hub location problem and multicommodity network design problem, to better

[1]The levels of load-imbalance and link-imbalance of 60% and under (Ψ and $\Theta \leq 0.6$) ensure that, in the worst case, the empty mileage from RP-network compares favorably (or better if Ψ and Θ are below 0.6) with industry averages. The percentage circuitry levels (Ω, $\bar{\Omega}^A$, and $\bar{\Omega}^M$) throughout this study are calculated very conservatively based on Euclidean distances. Thus, the actual levels would be lower if the shortest path distances are used (since we consider incomplete networks). The percentage circuitry levels can be further lowered if we consider direct shipment for the short distance commodities with large percentage circuitry levels.

represent real problems. Note that although the main objective is to construct the transmission network, our models can also be used for the purpose of upgrading and extending an existing network.

3. The solution algorithms developed for solving our four models can be applied to problems of a similar nature, ones that consider constructing networks and routing commodities simultaneously, such as hub location problems and multicommodity network design problems. An alternative approach to strengthen cuts is introduced; strengthened cuts are also applicable to different types of cuts. The ε-optimal framework is enhanced by the use of a local search. The copy constraints are defined in aggregated format. Different heuristics algorithms are developed for each applications.

VI.2. Foundation for Future Research

Future extensions of the models and solution algorithms developed in this dissertation will consider additional complexities and/or generalizations of the problems, as summarized below:

1. **Multiple assignment of nodes:** Throughout this study, the assumption of single assignment is made and every node can access the relay network only via a single relay point. Relaxing this assumption will generalize the model and permit multiple assignment of nodes. Thus, nodes can now access the relay network through multiple relay points and commodities will be transferred in the most direct direction. As a result, the total transportation distance, cost, and time could be reduced. Moreover, additional accessibility also improves network performance, especially in terms of reliability and survivability.

2. **Capacity on relay points:** In our study, capacity is defined between a

pair of relay points. In fact, capacity can also be explicitly defined on the relay points themselves (e.g., number of drivers at the relay points), as it is defined on hubs for capacitated hub location problems. The problem would become more constrained, and at the same time, be more generalized, as it could be shown that the arc-capacity can be transformed into node-capacity. Moreover, capacity limitation can be in the form of total flows or total number of connections.

3. **Unsplittable demand:** In all four models, demand could be split and transferred using multiple routes. However, demand can be unsplittable and require a single flow path (Barnhart et al., 2000), even under tight capacity limitation (e.g., teleconferencing). In this case, our Benders decomposition algorithms (as in Model 3) can effectively handle this requirement if the flow subproblem has integrality properties and decomposable structure. Otherwise, the model must be transformed or else, a different approach (Lagrangean approach as in Model 4 or other appoaches) may be more applicable.

4. **Different technology:** From the experimental results, we have observed that, in many case, link utilization can be very unevenly distributed. Since capacity requirement in different regions can vary significantly, having different link technologies with different capacities and fixed costs would help control construction budgets and make the problem more realistic. However, the solution space could be greatly enlarged and solving such a problem may require major modification. Moreover, exploring a step cost function for relay points and links with different capacity levels would also be an interesting research direction.

5. **Applying solution approaches to the other problem domains:** The relay network design problem is closely related to the hub location problems

and multicommodity network design problems. According to our earlier discussion, under some parameter settings, Models 1 and 2 are the same as the uncapacitated single assignment hub location problem, and Models 3 and 4 are essentially the uncapacitated and capacitated multicommodity network design problems. Additionally, our problem could be transformed into a single source facility location problem. Therefore, it would be interesting to observe the performance of our solution algorithms developed in this dissertation when applied to solving these classes of problems.

REFERENCES

Abdinnour-Helm, S., M. A. Venkataramanan. 1998. Solution approaches to hub location problems. *Annals of Operations Research* **78** 31–50.

Ahuja, R. K., T. L. Magnanti, J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Upper Saddle River, NJ.

Ali, T. H., S. Radhakrishnan, S. Pulat, N. C. Gaddipati. 2002. Relay network design in freight transportation systems. *Transportation Research Part E: Logistics and Transportation Review* **38**(6) 405–422.

Alumur, S., B. Y. Kara. 2008. Network hub location problems: The state of the art. *European Journal of Operational Research* **127**(1) 1–21.

Alvarez, A. M., J. L. González-Velarde, K. De-Alba. 2005a. Grasp embedded scatter search for the multicommodity capacitated network design problem. *Journal of Heuristics* **11**(3) 233–257.

Alvarez, A. M., J. L. González-Velarde, K. De-Alba. 2005b. Scatter search for network design problem. *Annals of Operations Research* **138**(1) 159–178.

American Trucking Association (ATA). 2006. Standard trucking and transportation statistics, (Vol.13, No.1). http://www.atabusinesssolutions.com/p-23-ata-standard-trucking-and-transportation-statistics-stats.aspx.

American Trucking Association (ATA). 2007. Trucking activity report, (Vol.15, No.3). http://www.atabusinesssolutions.com/p-24-ata-trucking-activity-report-trac.aspx.

American Trucking Association (ATA). 2009. US freight transportation forecast to 2020. http://www.atabusinesssolutions.com/p-209-ata-us-freight-transportation-forecast-to-2020.aspx.

Balakrishnan, A., T. L. Magnanti, P. Mirchandani. 1997. Network design. M. Dell' Amico, F. Maffioli, S. Martello, eds., *Annotated Bibliographies in Combinatorial Optimization*. Wiley, Chichester, UK, 311–334.

Balakrishnan, A., T. L. Magnanti, R. T. Wong. 1989. A dual-ascent procedure for large-scale uncapacitated network design. *Operations Research* **37**(5) 716–740.

Barnhart, C., C. A. Hane, P. H. Vance. 2000. Using branch-and-price-and-cut to solve origin-destination integer multicommodity flwo problems. *Operations Research* **48** 318–326.

Belotti, P., F. Malucelli, L. Brunetta. 2007. Multicommodity network design with discrete node costs. *Networks* **49**(1) 90–99.

Benders, J. F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4** 238–252.

Birge, J. R., F. Louveaux. 1998. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research* **34** 384–392.

Bureau of Economic Analysis. 2006. Industry economic accounts. http://www.bea.gov/industry/gpotables.

Bureau of Labor Statistics. 2007. Job opening and labor turnover survey. http://www.bls.gov/jlt/.

Cabral, E. A., E. Erkut, G. Laporte, R. A. Patterson. 2007. The network design problem with relays. *European Journal of Operational Research* **180**(2) 834–844.

Cabral, E. A., E. Erkut, G. Laporte, R. A. Patterson. 2008. Wide area telecommunicatino network design: Application to the alberta supernet. *Journal of the Operational Research Society* **59**(11) 1460–1470.

Campbell, J. F. 1994. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research* **72** 387–405.

Campbell, J. F., A. T. Ernst, M. Krishnamoorthy. 2002. Hub location problems. Z. Drezner, H. W. Hamacher, eds., *Facility Location: Applications and Theory*. Springer-Verlag, New York, 373–407.

Campbell, J. F., A. T. Ernst, M. Krishnamoorthy. 2005a. Hub arc location problems: Part i-introduction and results. *Management Science* **51**(10) 1540–1555.

Campbell, J. F., A. T. Ernst, M. Krishnamoorthy. 2005b. Hub arc location problems: Part ii-formulations and optimal algorithms. *Management Science* **51**(10) 1556–1571.

Contreras, I., J. A. Déaz, E. Fernández. 2008. Lagrangean relaxation for the capacitated hub location problem with single assignment. *European Journal of Operational Research* **184** 468–479.

Costa, A. M. 2005. A survey on benders decomposition applied to fixed-charge network design problem. *Computers & Operations Research* **32** 1429–1450.

Crainic, T. G., A. Frangioni, B. Gendron. 2001. Bundle-based relaxation methods for multicommodity capacitated fixed charge network design. *Discrete Applied Mathematics* **112**(1-3) 73–99.

Crainic, T. G., M. Gendreau. 2002. Cooperative parallel tabu search for capacitated network design. *Journal of Heuristics* **8**(6) 601–627.

Crainic, T. G., M. Gendreau, J. M. Farvolden. 2000. A simplex-based tabu search method for capacitated network design. *Journal on Computing* **12**(3) 223–236.

Crainic, T. G., B. Gendron, G. Hernu. 2004. A slope scaling/lagrangean perturbation heuristic with long-term memory for multicommodity capacitated fixed-charge network design. *Journal of Heuristics* **10**(5) 525–545.

Crainic, T. G., Y Li, M. Toulouse. 2006. A first multilevel cooperative algorithm for capacitated multicommodity network design. *Computers and Operations Research* **33**(9) 2602–2622.

De Camargo, R. S., G. Miranda, H. P. Luna. 2008. Benders decomposition for the uncapacitated multiple allocation hub location problem. *Computers and Operations Research* **35** 1047–1064.

Deloitte Touche Tohmatsu. 2009. Telecommunications predictions TMT trends 2009. http://www.deloitte.com/view/en_BA/ba/industries/technology-media-and-telecommunications/article/485163ec49101210VgnVCM100000ba42f00aRCRD.htm.

Ebery, J., M. Krishnamoorthy, A. Ernst, N. Boland. 2000. The capacitated multiple allocation hub location problem: Formulations and algorithms. *European Journal of Operational Research* **120**(3) 614–631.

Erlbaum, N., J Holguín-Veras. 2006. Some suggestions for improving cfs data products. *Transportation Research E-Circular* **1** 77–97. http://onlinepubs.trb.org/onlinepubs/circulars/ec088.pdf.

Ernst, A. T., Hamacher H., H. Jiang, M. Krishnamoorthy, Woeginger G. 2009. Uncapacitated single and multiple allocation p-hubs center problems. *Computers & Operations Research* **36** 2230–2241.

Ernst, A. T., M. Krishnamoorthy. 1996. Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location Science* **4**(3) 139–154.

Ernst, A. T., M. Krishnamoorthy. 1999. Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research* **86** 141–159.

Gendron, B., T. G. Crainic. 1994. Relaxation for multicommodity capacitated network design problems. Publication CRT-965, Centre de recherch sur les transports, University de Montreal, Canada.

Gendron, B., T. G. Crainic, A. Frangioni. 1998. Multicommodity capacitated network design. B. Sansó, P. Soriano, eds., *Telecommunications Network Planning*. Kluwer, Berlin 1–19.

Geoffrion, A. M., G. W. Graves. 1974. Multicommodity distribution system design by Benders decomposition. *Management Science* **20**(5) 822–844.

Ghamlouche, I., T. G. Crainic, M. Gendreau. 2003. Cycle-based neighborhoods for fixed-charge capacitated multicommodity network design. *Operations Research* **51**(4) 655–667.

Ghamlouche, I., T. G. Crainic, M. Gendreau. 2004. Path relinking, cycle-based neighborhoods, and capacitated multicommodity network design. *Annals of Operation Research* **131** 109–133.

Guignard, M., S. Kim. 1987. Lagrangean decomposition: A model yielding stronger lagranfean bounds. *Mathematical Programming* **39** 215–228.

Holmberg, K., J. Hellstrand. 1998. Solving the uncapacitated network design problem by a lagrangian heuristic and branch-and-bound. *Operations Research* **46**(2) 247–259.

Holmberg, K., D. Yuan. 2000. A lagrangian heuristic based branch-and-bound approach for the capacitated network design problem. *Operations Research* **48**(3) 461–481.

Hunt, G. W. 1998. Transportation relay network design. Ph.D. dissertation, Georgia Institute of Technology, Atlanta, GA.

Kashyap, A., F. Sun, M. Shayman. 2006. Relay placement for minimizing congestion in wireless backbone networks. *Wireless Communications and Networking Conference,* Las Vegas, NV.

Keller, S. B. 2002. Driver relationship with customers and driver turnover: Key mediating variables affecting driver performance in the field. *Journal of Business Logistics* **23** 39–65.

Keller, S. B., J. Ozment. 1999. Managing driver retention: Effects of the dispatcher. *Journal of Business Logistics* **20** 97–120.

Klincewicz, J. G. 1996. A dual algorithm for the uncapacitated hub location problem. *Location Science* **4**(3) 173–184.

Magnanti, T. L., P. Mireault, R. T. Wong. 1986. Tailoring Benders decomposition for uncapacitated network design. *Mathematical Programming Study* **26** 112–154.

Magnanti, T. L., R. T. Wong. 1981. Accelerating Benders decomposition: Algorithmic enhancement and model selection. *Operation Research* **29**(3) 464–484.

Marin, A. 2005. Formulating and solving splittable capacitated multiple allocation hub location problems. *Computers and Operations Research* **32**(12) 3093–3109.

Mayer, G., B. Wagner. 2002. HubLocator: An exact solution method for the multiple allocation hub location problem. *Computers and Operations Research* **29**(6) 175–739.

Mele, J. 1989a. Carriers cope with driver shortage. *Fleet Owner* **84**(1) 104–111.

Mele, J. 1989b. Solving driver turnover. *Fleet Owner* **84**(9) 45–52.

Min, H., A Emam. 2003. Developing the profiles of truck dreivers for their successful recruitment and retention. *International Journal of Physical Distribution & Logistics Management* **33** 149–162.

Min, H., T. Lambert. 2002. Truck driver shortage revisited. *Transportation Journal* **42** 5–17.

O'Kelly, M. E., H. J. Miller. 1994. The hub network design problem. *Journal of Transport Geography* **2**(1) 31–40.

Pedersen, M. B., T. C. Crainic, O. B. G. Madsen. 2009. Model and tabu search metaheuristics for service network design with asset-balance requirement. *Transportation Science* **43**(2) 158–177.

Pirkul, H., D. A. Schilling. 1998. An efficient procedure for designing single allocation hub and spoke systems. *Management Science* **44**(12) 235–242.

Randall, M. 2008. Solution approaches for the capacitated single allocation hub location problem using ant colony optimisation. *Journal Computational Optimization and Applications* **39**(2) 239–261.

Rardin, R. L., U. Choe. 1979. Tighter relaxations of fixed charge network flow problem. Tech. Rep. J-79-18 School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Richardson, H. L. 1994. Can we afford the driver shortage. *Transportation and Distribution,* **35**(8) 30–34.

Rodriguez, J., M. Kosir, B. Lantz, G. Griffin, J. Glatt. 2000. The costs of truckload driver turnover. Tech. Rep. Report No. SP-146 Upper Great Plains Transportation Institute, North Dakota State University, ND. http://www.ugpti.org/pubs/pdf/SP146.pdf.

Rodríguez-Martín, I., J. J. Salazar-González. 2008. Solving a capacitated hub location problem. *European Journal of Operational Research* **184** 468–479.

Sarkar, S., H. H. Yen, S. Dixit, B. Mukherjee. 2009. Hybrid wireless-optical broadband access network (woban): Network planning using lagrangean relaxation. *IEEE/ACM Transactions on Networking* **17**(4) 1094–1105.

Schneider Logistics Inc. 2009. 2008-2009 State of the Industry Review. http://www.schneider.com/www1/groups/public/@marketing-public-general/documents/document/prd_002492.pdf.

Silva, M. R., Cunha C. B. 2009. New simple and efficient heuristics for the capacitated single allocation hub location problem. *Computer & Operations Research* **36** 3152–3165.

So, A., B Liang. 2006. Optimal placement of relay infrastructure in heterogeneous wireless mesh networks by benders decomposition. *Proceedings of the 3rd International Conference on Quality of Service in Heterogeneous Wired/wireless Networks* Waterloo, Ontario, Canada.

Taha, T. T., G. D. Taylor. 1994. An integrated modeling framework for evaluating

hub-and-spoke networks in truckload trucking. *Logistics and Transportation Review* **30**(2) 141–166.

Taylor, G. D., S. Harit, J. R. English, G. L. Whicker. 1995. Hub and spoke networks in truckload trucking: Configuration testing and operational concerns. *Logistics and Transportation Review* **31**(3) 209–237.

Taylor, G. D., T. S. Meinert. 2000. Improving the quality of operations in truckload trucking. *IIE Transactions* **32**(6) 551–562.

Taylor, G. D., T. S. Meinert, R. C. Killian, G. L. Whicker. 1999. Development and analysis of alternative dispatching methods in truckload trucking. *Transportation Research Part E* **35**(3) 191–205.

Taylor, G. D., G. L. Whicker, J. S. Usher. 2001. Multi-zone dispatching in truckload trucking. *Transportation Research Part E* **37**(5) 375–390.

Transportation Topics. 2007. June 25. https://www.cambeywest.com.

Transportation Topics. 2008. September 22. https://www.cambeywest.com.

Truckload Carriers Association. 2004. How to recruit and retain drivers. http://www.commercialcarrieruniversity.com/ccu_driver_recruitment.shtml.

US Census Bureau, US. Department of Commerce. 2006. Service annual survey. http://www2.census.gov/services/sas/data/Historical/sas-06.pdf.

US Census Bureau, US. Department of Commerce. 2008. 2007 commodity flow survey. http://www.bts.gov/publications/commodity_flow_survey.

US Department of Transportation. 2008. Freight facts and figures 2008. http://ops.fhwa.dot.gov/freight/freight_analysis/nat_freight_stats/docs/08factsfigures.

Üster, H., G. Easwaran, E. Akçalı, S. Çetinkaya. 2007. Benders decomposition with alternative multiple cuts for a multi-product closed-loop supply chain network design model. *Naval Research Logistics* **54** 890–907.

Üster, H., N. Maheshwari. 2007. Strategic network design for multi-zone truckload shipments. *IIE Transactions* **39**(2) 177–189.

Vacca, J. R. 2001. *High-Speed Cisco Networks: Planning, Design, and Implementation*. Auerbach, Boca Raton, FL.

Wagner, B. 2007. An exact solution procedure for a cluster hub location problem. *European Journal of Operational Research* **178**(2) 391–401.

Wagner, B. 2008. Model formulations for hub covering problems. *Journal of the Operational Research Society* **59** 932–938.

Yoon, M. G., J. Current. 2008. The hub location and network design problem with fixed and variable arc costs: Formulation and dual-based solution heuristic. *Journal of the Operational Research Society* **59** 80–89.

VITA

Panitan Kewcharoenwong received his Bachelor of Engineering degree in industrial engineering from Sirindhorn International Institute of Technology, Thailand, in 2002 and Master of Arts in economics from Chulalongkorn University, Thailand, in 2003. He joined the Department of Industrial and Systems Engineering at Texas A&M University in Fall 2004 for doctoral studies and graduated with his Ph.D. in May 2010. His research interests are in the development of mathematical models for transportation logistics and telecommunications applications, and efficient methodologies to solve these models.

Panitan Kewcharoenwong can be reached at:
501 Eakpailin Vg., Srinakarin Rd.
Bangkaew, Bangpli
Samutprakarn, 10540
Thailand