SCHEDULING TRANSSHIPMENT OPERATIONS IN MARITIME CHEMICAL TRANSPORTATION

HUANG CHENG

(*B.Eng.* (*Hons*.) *NUS*)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF ENGINEERING

DEPARTMENT OF CHEMICAL AND BIOMOLECULAR ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2005

ACKNOWLEDGEMENTS

I would like to extend my sincerest gratitude and deepest appreciation to the following people for their help and support throughout the course of study.

A/P I. A. Karimi, my research supervisor and constant source of support, guidance, encouragement and inspiration.

Arul, Ganesh, Li Jie, Liu Yu, Mukta, Suresh and Wenkai, for their help and willingness in sharing their valuable knowledge and personal experiences with me.

Last but not least, all my friends for their help in one way or another during the study.

TABLE OF CONTENTS

SUMMARY

In this work, we address two scheduling problems of chemical transshipment activities in chemical logistics. The first one focuses on general transshipment operation, the second one aims to solve tanker lightering operation.

Maritime logistics, in spite of its key role in global chemical supply chains, has received little attention in the literature. Transshipment or direct ship-to-ship transfer of bulk liquid cargos for regional distribution is a common activity in maritime chemical logistics. Efficient scheduling of transshipment operations is economically crucial because of the high costs of the participant multi-parcel carriers. In the first part of research, we consider a general chemical transshipment problem, in which multiple donor carriers transship several chemical cargos to multiple recipient carriers. We develop nine continuous-time mixed-integer linear programming (MILP) formulations of three types for determining the optimal sequences, positions, and timings for unloading various cargos. Two of these models using the big-M relaxation seem the best in solving several test problems involving up to sixteen cargos with reasonable speed, but their performances vary with the numbers of two-sided cargos and ships. For even larger problems, we present a cargo aggregation heuristic that tremendously speeds up the solution of our models, yet gives near-optimal solutions. The research finding also suggests that the slot-based sequencing approach is generally more efficient than the pair-wise approach in many scheduling problems.

In the second part of research, we address a tanker lightering problem. Lightering vessels receive crude oil from crude tankers in order to reduce their drafts. Then, crude tankers can deliver crudes to shallow refinery ports where they cannot enter previously due to draft limitation. Good lightering schedules are of utmost importance to enhance the efficiency of lightering operation and reduce the total operating cost. We develop a continuous-time MILP formulation using two alternate linearization methods to address two different objectives. One objective considers the time-charter cost of tankers, while the other considers the demurrage cost of tankers. Our formulations include many practical and important features. For example, we consider multi-compartments vessels, restrict the number of simultaneous transfers for a single tanker to two, consider the differences in crude densities, and allow the freedom to select lightering crudes. In contrast to the general chemical transshipment problem, the volumes and assignments to lightering vessels in this case are decided by the optimization model. In addition, the system cost here is an indicator of the customer satisfaction level as well as the utilization of fleet of lightering vessels. Our MILP models generate optimal lightering schedule with lightering volumes, sequence, times, and assignments, which minimizes the operating costs of lightering vessels, the demurrage or time-charter costs of tankers as well as the delivery times of crude oil from the lightering location to refinery ports. We also develop some heuristic methods to simplify the models when addressing large size problems. Furthermore, the comparison between our slot-based model and event-based model from literature also shows the superiority of slot-based approach over event-based approach in lightering scheduling problems.

NOMENCLATURE

SYMBOLS

Chapters 3-4

Indices

i	real transshipment cargos
j	one-sided cargos
k	time slots
n	multi-compartment short-sea carriers
m	multi-compartment deep-sea carriers
p	position slots

Sets

$\boldsymbol{J}$	Set of one-sided cargos derived from the original I transshipment cargos
$\boldsymbol{TSC}$	Set of two-sided cargos
$\boldsymbol{OSC}$	Set of one-sided cargos
$\boldsymbol{JS}_n$	Set of cargos j for small ship n
$\boldsymbol{JB}_m$	Set of cargos j on big ship m
$\boldsymbol{J}_p$	Set of cargos j that must transfer from position p
$\boldsymbol{IS}_n$	Set of actual cargos i for short-sea carrier n
$\boldsymbol{P}_m$	Set of positions p that belong to big ship m
$\boldsymbol{P}_n$	Set of small ships n that receive cargos from position p

Parameters

$ETAS_n$	Estimated time of arrival for short-sea carrier n
$ETAB_m$	Estimated time of arrival for deep-sea carrier m
$ETAC_j$	Time at which cargo j becomes available for transshipment
τ_j	Service time required to transfer cargo j

$\theta_{npp'}$	Travel time required by small ship n to travel from position p to p'
$TCCB_m$	Time-charter cost of deep-sea carrier m
$TCCS_n$	Time-charter cost of short-sea carrier n
MT_j	Minimum start time of cargo j
MTB_p	Minimum start time at position p
MTS_n	Minimum start time on small ship n
K_n	Number of slots on small ship n
K_p	Number of slots on position p
K_m	Number of slots on big ship m
K_j	Number of destination slots of cargo j
H	Some large number

Variables

x_{jk}	1, if cargo j transfers during slot k
U_j	1, if cargo j transfers
TS_{nk}	Start time of slot k on small ship n
z_{npk}	1, if small ship n is at position p during slot k
$Z_{npp'k}$	1, if small ship n moves from position p to p' during slot k
$y_{jj'}$	1, if cargo j' transfers later than cargo j
T_j	Time at which cargo j transfers
TX_{jk}	$= TS_{nk}x_{jk} = T_jx_{jk}$ for F1 models and $= TB_{pk}x_{jk} = T_jx_{jk}$ for F2 and F3 models
DTB_m	Departure time of deep-sea carrier m
DTS_n	Departure time of short-sea carrier n
$TTCC$	Total time-charter cost of all ships
TB_{pk}	Start time of slot k at position p
$d_{jj'j''}$	0-1 continuous dummy variable

$\delta_{jj'}$	0-1 continuous dummy variable

Chapters 5-6

Indices

m	crude tankers
n	lightering vessels
r	refineries
p	position slots
k	time slots
c	crudes
j	lightering parcels

Sets

$\boldsymbol{CT}_m$	Set of crudes c that tanker m carries
$\boldsymbol{CV}_n$	Set of crudes c that can be lightered by lightering vessel n
$\boldsymbol{S2}$	Set of stage two parcels of 2-stage tankers
$\boldsymbol{JC}_c$	Set of parcel j that has crude oil c
$\boldsymbol{JB}_m$	Set of parcel j that is from tanker m
$\boldsymbol{J}_p$	Set of parcel j that transfers from position p for $m(j) \in \boldsymbol{LT}$
$\boldsymbol{JS}_n$	Set of parcel j that can be unloaded by lightering vessel n
$\boldsymbol{JJ}$	Set of parcel pairs that can be unloaded by the same lightering vessel during the same voyage
$\boldsymbol{TST}$	Set of tankers that their physical tankers are 2-stage tankers
$\boldsymbol{MM}$	Set of tanker pairs that are offshore and anchorage tankers respectively of the same 2-stage physical tanker
$\boldsymbol{LT}$	Set of large tankers that require more than two parcels for lightering

Parameters

PS_c	Volume of crude oil c (m^3)
LW_m	Lightering weight of tanker m (kg)
N_n^U	Number of compartments of lightering vessel n
SC_n	Size of compartments of lightering vessel n (m^3)
vf_n	Velocity of lightering vessel n when loaded
ve_n	Velocity of lightering vessel n when empty

FIN_{nc}	Volumetric pumping rate at which lightering vessel *n* receives crude *c* from tanker
$FOUT_n$	Volumetric pumping rate at which lightering vessel *n* discharges crude to refinery
$ETAS_n$	Expected time of arrival of lightering vessel *n*
$ETAB_m$	Expected time of arrival of tanker *m*
ETA_j	Expected time of arrival of parcel *j*
ρ_j	Density of parcel *j*
dr_j	Distance of parcel *j* to destination refinery
da_j	Distance of parcel *j* from anchorage
WD_{nj}	Maximum weight of parcel *j* that vessel *n* can carry
VC_n	Fixed operating cost per voyage of vessel *n* (voyage cost)
FC_n	Fuel cost per nm of lightering vessel *n*
MDT_j	Mounting and dismounting time needed for parcel *j*
DTR_j	Docking and undocking time at a refinery needed for parcel *j*
NS_m^U	Maximum number of parcels needed for tanker *m*
NS_m^L	Minimum number of parcels needed for tanker *m*
WT_n^U	$= \max_{j \in \boldsymbol{JS}_n} WD_{nj}$
VT_n^U	$= \max_{j \in \boldsymbol{JS}_n} \frac{WD_{nj}}{\rho_j}$
DD_j	Due date of parcel *j*
DDP_j	Due date penalty of parcel *j*
TCC_m	Time-charter cost of tanker *m*
AD_m	Agreed duration of tanker *m*
DC_m	Demurrage cost of tanker *m*
TOA_m	Travel time from offshore to anchorage of tanker *m*
$TPP_{njj'}$	Time for vessel *n* to travel from parcel *j* to *j*' directly
TPR_{nj}	Time for vessel *n* to take parcel *j* to its destination refinery
TRA_{nj}	Time for vessel *n* to travel from destination refinery of parcel *j* to anchorage
TAP_{nj}	Time for vessel *n* to travel from anchorage to pick up parcel *j*
K_n	Number of time slots on lightering vessel *n*

HT	Big-M constant for time constraint

Variables

x_{nkj}	1, if vessel n transfers parcel j during slot k
$y_{jj'}$	1, if parcel j' is lightered sometime after parcel j
ze_{nk}	1, if vessel n ends its current voyage in slot k
U_{nk}	1, if vessel n transfers a parcel during slot k
$Z_{nkjj'}$	1, if vessel n transfers parcel j in slot k, j' in slot k+1 during the same voyage
WT_{nk}	Total weight of crudes collected by vessel n up to and including slot k on the current voyage
WZ_{nk}	$= WT_{nk}(1 - ze_{nk})$
VP_{nkj}	Volume of parcel j that vessel n withdraws during slot k
VT_{nk}	Total volume of crudes collected by vessel n up to and including slot k on the current voyage
VZ_{nk}	$= VT_{nk}(1 - ze_{nk})$
NC_{nk}	Integer variable, number of compartments used during slot k of vessel n
NT_{nk}	Total number of compartments used by vessel n up to and including slot k on the current voyage
NZ_{nk}	$= NT_{nk}(1 - ze_{nk})$
TS_{nk}	Time when lightering vessel n starts slot k
T_j	Time when at which parcel j transfers
DTB_m	Departure time from lightering location (offshore or anchorage) of tanker m
TD_n	Total traveling distance of vessel n
ATR_{nkj}	Arrival time of a parcel j of a small tanker at its destination refinery via vessel n during slot k
ATR_j	Arrival time of a parcel j of a large tanker at the refinery
$d_{jj'j''}$	0-1 continuous dummy variable
$\delta_{jj'}$	0-1 continuous dummy variable
TC	Total cost of the entire system

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Logistics in Chemical and Petrochemical Industry

The chemical and petrochemical industry is highly global with worldwide procurement and distribution. In today's business environment, companies are also free to locate their businesses all over the world to gain maximum benefits. For example, a petrochemical company produces/imports crude oil from the major oil fields located in Middle East; refines crude oil in Southeast Asia countries because the local labor and operating cost are cheap, sells products to downstream manufacturers and end customers all over the world. By doing this, companies can utilize their resources, minimize costs hence boost up profits. Therefore, large quantities of liquid chemicals (raw materials, intermediate products, and finished products) must move globally over long distances (e.g. from Middle East to Southeast Asia, from Southeast Asia to Europe and USA, and so on). Of course, this costs money, and logistics costs can be as high as 20% in chemical supply chains as highlighted by Karimi et al. (2002) and efficient logistics is the key to successful supply chains.

1.2 Shipping in Chemical Logistics

There are many ways of transporting chemical cargos including air, truck, rail, ship, pipeline, etc. Although pipelines are preferred, whenever possible, ocean shipping is the dominant transport mode, when moving large volumes of bulk liquid chemicals between continents. It is more economical than air, truck, and rail, and is more flexible than pipeline. 65 to 85 weight percent (wt%) of international trade is via sea transport, which clearly indicates the monopoly of ocean shipping (Christiansen et al., 2003). In addition to the seaborne trade, shipping is also very important in domestic trade for countries with long shorelines, or navigable waterways and rivers, or many islands, such as, Indonesia, Norway, Denmark, USA, Philippines and so on. Chemical logistics is no exception.

There are three basic modes of operation of shipping, *industrial*, *liner* and *tramp* shipping (Lawrence, 1972). For *industrial* shipping, the companies own the cargos transferred and control the ships used to transport them. The companies own or charter the ships. The objective for the company is to minimize the total operating cost while attaining the delivery requirements of cargos. For *liners*, they normally operate according to published routes or schedules. It is similar to bus service. And they normally travel long, intercontinental routes. The objective is to maximize the ships utilization hence to gain maximum profits. For *tramp* ships, usually they are engaged in small contracts, and they pick up available/profitable cargos. It is similar to taxicab service. The objective for the shipping company is to maximize the profit. Cargo delivery may require more than one mode of operation. For example, a cargo is transported from one major port in Europe to another major port in Southeast Asia by a liner. But, the ultimate destination for this cargo is a port that is not in the liner's published route. As a result, further transport by a tramp (charted) ship is required. Transfer of cargo from one ship to another ship is required in such practice. This operation can be direct or indirect (delayed) depending on the type of cargos transferred, the availability of recipient ships and other business considerations. We will elaborate on this operation in detail later.

In chemical shipping where liquid cargos are normally shipped, there is another way of categorizing shipping operations. We classify chemical shipping into two types, deep-sea shipping and short-sea shipping. Jetlund and Karimi (2004) defined deep-sea shipping as the transportation between continents in deep-sea water, where large multi-compartment tankers transport large amounts of cargos between major ports and manufacturers. Short-sea shipping focuses on regional areas. It normally involves smaller, multi-compartment vessels that travel relatively short distances between regional ports. Deep-sea shipping acts as a feeder to short-sea shipping. When deep-sea carriers arrive at major ports, they not only unload some cargos, but they also directly (ship-to-ship) transfer some cargos to short-sea carriers for further delivery to regional ports. This reduces transport costs, because the fuel and time-charter costs of deep-sea carriers are far greater than those of short-sea carriers. Furthermore, deep-sea carriers often cannot enter shallow destination ports due to draft limitations. Then, the only way to deliver cargos to the destinations is by transferring them to the smaller carriers that can access the ports. The operation of transferring cargos directly (ship-to-ship) from intercontinental deep-sea carriers (liners) to regional short-sea carriers (tramp ships) or in general from one vessel to another is called transshipment.

1.3 Transshipment in Chemical Shipping

1.3.1 Main feature of transshipment operation

The main feature that distinguishes the transshipment of chemicals from that of other goods is that the transfer must be direct via a hose, making it necessary for both the donor and recipient ships to be engaged in the operation simultaneously. Unlike most other goods or containers that can simply be stored at a port for a period before another ship collects them, a donor ship cannot simply dump a non-containerized chemical cargo at a port and leave, and let the recipient ship collect it some time later. Most ports do not have facilities for such temporary storage. Such a delayed transfer would normally require a third party logistics (3PL) facility and would incur significant additional cost. Such feature makes it difficult to arrange the transshipment operations in an efficient manner, whereas, a good schedule is the key to reduce logistics cost.

1.3.2 Need for transshipment operations

Transshipment operation is very common practice in the chemical shipping industry. Christiansen et al. (2003) showed that the seaborne trade has been increasing during the last decade, and is expected to grow further with the enlarging world economy. Furthermore, Ng and Baumgarten (1998) reported that the trade in chemical and petrochemical industries within regional areas is also increasing in recent years. Thus, there is a growing need for supplying and distributing chemicals to and from industries within regions that are uneconomical for the inter-continental or deep-sea routes to serve. As deep-sea and short-sea shipping activities increase, so do the transshipment activities between them.

In addition, the myriad of mergers, acquisitions, and collaboration are also increasing the demand for transshipment operations. The major players in the ocean shipping industry normally operate a fleet of deep-sea carriers and focus their businesses on deep-sea trading. To capture the growing demand for regional transportation of liquid chemicals, more and more shipping companies are allying and collaborating with regional shipping players. This is to expand their operational reach, to gain greater flexibility in offering services to global markets as well as regional markets, to increase profits by reducing operating costs, and to enhance the fleet utilization as explained by Sheppard and Seidman (2001). This collaboration allows the chemical shipping companies to integrate their global services with regional services, to aggregate the cargos from deep-sea carriers, and to redistribute and deliver them to regional ports; in which transshipment operations are required.

1.3.3 Special case: tanker lightering

A tanker is one of the most common and economical modes for transporting large quantities of liquid cargos, especially crude oil. In year 2003, 1,686 million tons of crude oil was shipped worldwide (UNCTAD, 2004). However, fully loaded large tankers cannot pass through shallow channels or dock at shallow ports due to shallow draft, narrow entrance, or small berth. Under such circumstances, small vessels are often employed to unload a part of the crude oil from the tanker to reduce its draft and enable its entry into a shallow channel or port. Subsequently, the tanker and the small vessels both travel to the refinery port to deliver the crude oil. This operation of transferring crude oil from large tankers to small vessels in order to lighter the tankers is called tanker lightering and the small vessels are called lightering vessels.

The tanker lightering operation is a part of the petroleum supply chain. Though it requires additional cost, it offers several advantages to a refinery. Firstly, as described earlier, many refinery ports have shallow draft and large tankers are the only economical means to deliver crude oils to them. Secondly, tanker lightering helps reduce the demurrage of tankers by reducing their waiting time for unloading and reducing the refinery inventory holding costs by ensuring on-time delivery of crude oils (Chajakis, 2000). Furthermore, it adds flexibility to crude oil supplies. For instance, faster delivery is possible by discharging crudes to multiple storage tanks from multiple vessels simultaneously, and delivering a part of the crudes to refineries that need them urgently (Lin et al., 2003).

1.4 Scheduling in Transshipment Operations

Typical transshipment operations proceed as follows. The carriers (deep-sea and short-sea) arrive at a transshipment location at some estimated times. Since multiple ships may be involved in a transshipment operation and multiple transshipment operations may overlap in time, queues of ships may develop and congestion may arise. This congestion may lead to delays and subsequent costs, if one does not synchronize and schedule the various requests optimally. Clearly, a careful scheduling is crucial and extremely important under such circumstances for the shipping companies, as ships are highly capital-intensive assets. The time-charter or total operating cost of a multi-parcel chemical tanker can be several tens of thousands of US dollars per day. Besides, port costs also increase with the time that a ship spends at a port and can be substantial. Sometimes, even demurrage cost of tankers may be important and this can be several thousand US dollars per day. Therefore, there is a tremendous need for systematic scheduling procedures that minimize the total cost of a transshipment operation.

The same applies to tanker lightering operation, too. The demurrage costs of tankers are extremely high. During congestion, tankers may easily spend days awaiting lightering service. Therefore, effective scheduling of lightering operation is also crucial for minimizing the system cost by reducing the waiting times of tankers and increasing the utilization of lightering vessels.

1.5 Research Objectives

This research work aims to develop several mathematical models to help chemical shipping companies to generate short-term optimal schedules to increase the efficiency and reduce the operating cost of transshipment operations; where bulk liquid chemical cargos are directly transferred from deep-sea, multi-parcel, bulk chemical carriers to short-sea, multi-parcel carriers for further regional distribution. In addition, the research work also addresses the special case of transshipment operation: tanker lightering; where mathematical models are developed to schedule tanker lightering operations efficiently.

There are many possible mathematical models to solve the same problem. But, only some of them are effective and efficient. Hence, we develop several alternate models all using continuous-time representations and mixed integer linear programming (MILP) techniques; but using different modeling approaches and linearization methods. We will investigate the different factors affecting model performance by testing them over different types of examples. Then, we will suggest some guidelines for selecting a suitable model to use when facing different transshipment or tanker lightering problems. This is the second objective of our research work.

Moreover, in reality, practical problems are often very complicated and may involve thousands of parameters. In such case, generating optimal schedule may take days or even weeks, which is impractical for swift and prompt decisions. Therefore, our research work also includes developing heuristic methods to generate good transshipment schedules and tanker lightering schedules for large size problems.

1.6 Outline of the Thesis

In the next chapter (Chapter 2), we review the literature on general maritime shipping problems, including the previous work done on similar chemical transshipment problem and tanker lightering problem.

The remaining of the thesis consists of two major parts. The first part (Chapters 3-4) focuses on the scheduling of general transshipment operations; the second part (Chapters 5-6) focuses on the tanker lightering problem.

In Chapter 3, we propose three alternate and novel continuous-time MILP formulations. And each of them has three alternate formulations using different linearization methods. Then, Chapter 4 presents several examples to illustrate the application of our models, as well as to explore the factors affecting the model performance. We also propose some heuristics to simplify the models for larger problems.

In Chapter 5, we develop two alternate MILP formulations to address tanker lightering problem. And we identify two important objectives for the problem. In Chapter 6, we study the application and highlight the features of the models using several examples. In addition, we also demonstrate the advantages of our slot-based models over event-based models from the literature.

In Chapter 7, we conclude our research findings and provide recommendations for model extensions.

CHAPTER 2

LITERATURE SURVEY

The optimization problems of chemical shipping in maritime transportation can be classified into two types according to the length of planning horizon: planning and scheduling.

2.1 Planning in Maritime Transportation

The planning problem is a high level problem that considers long-term decisions, such as fleet size and mix, transportation system design, maritime supply chain design, and so on (Christiansen et al., 2003).

Fleet size and mix problem refers to the design of optimal fleet for a shipping company. This includes selecting optimal combination of types of ships, number of ships and sizes of ships in the fleet to maximize profit/minimize both operating and fixed costs. Dantzig and Fulkerson (1954) first considered a simple problem by assuming that all the ships in a fleet are of the same type, size and cost. The cargos are homogeneous with common loading/unloading port. They used integer variables in the model to determine the minimum number of tankers required for fulfilling the transportation requirement. A more practical problem considering heterogeneous fleet size and mix was studied by Fagerholt and Lindstad (2000). They proposed an algorithm to find an optimal routing policy, such as which vessels to operate, their weekly schedules, and so on.

The second decision involves the design of liner routes, frequencies of visiting major ports, and so on. The goal is to attract more contracts and enhance customer service levels while gaining maximum profit and increasing the fleet utilization. Rana and Vickson (1988) first addressed a liner routing problem for one ship. The same authors (1991) then extended the research to design liner routing for a fleet of ships.

The proposed model is a mixed integer nonlinear programming (MINLP) problem. They also proposed a solution algorithm using Lagrangean relaxation and decomposition method.

The third type of planning problem is to design maritime supply chain. Mehrez et al. (1995) addressed an industrial ocean-cargo shipping problem. They proposed a MILP model to help the company to select number and sizes of ships to use, and routes to follow for different phases in transporting cargos from original port to end customer. Cheng and Duran (2003) considered a worldwide crude oil transportation problem. They developed a decision support system to assist in deciding optimal sizes and types of tankers of a fleet as well as tanker routes in the worldwide crude oil supply chain. This decision support system uses both discrete event simulation and optimal control of the combined inventory and transportation system.

2.2 Ship Routing and Scheduling

The routing and scheduling problem involves medium-term decisions. The problem considerations vary from full to partial cargos, one to multiple cargos, fixed to variable cargo size, one to multiple products, compatible to non-compatible products and so on.

Brown et al. (1987) presented a scheduling problem of fully loaded crude oil for industrial shipping operations. Later, Fisher and Rosenwein (1989) discussed a similar problem with partial/full cargos. They considered a fleet of ships that are engaged in pickup and delivery of bulk cargos. The problem was solved optimally using interactive optimization system. Bausch et al. (1998) addressed a scheduling problem of multiple bulk products. The fleet of coastal tankers and barges transports multiple products in a load among plants, distribution centers and customers. They developed an optimization-based decision support system. In the first phase, every feasible schedule was generated using a detailed simulation. In the second phase, optimal schedule for the entire fleet with minimum operating cost was selected by an integer linear set partitioning model.

Fagerholt and Christiansen (2000a) studied a bulk ship scheduling problem combined with cargo allocation problem. The model decides cargo pickup and delivery with time windows, partition of ship's flexible holds, as well as allocation of multiple, compatible, fixed sized cargos. They used a set partitioning technique and was able to obtain optimal solutions for several case studies. The same authors (2000b) presented a traveling salesman problem with allocation, time window, and precedence constraints for ship scheduling in another work. The allocation constraints ensure the feasibilities of cargo allocations and ship's cargo holds partitions throughout the schedule. They employed a forward dynamic programming algorithm. Fagerholt (2001) studied a real ship scheduling problem with soft time windows. The problem is a multi-ship pickup and delivery problem. The soft time windows consideration allows violation of time window for some customers, hence, possible better schedules. They developed an optimization based approach suing set partitioning formulation to solve the problem.

Recently, Hwang et al. (2002) proposed a model for the routing and scheduling of a heterogeneous fleet. The fleet of ships is engaged in pickup and delivery of multiple bulk cargos with inventory constrained time windows. The industrial decisions involve sizes of cargos to carry, allocation to compartments and which ship to carry. They developed a non-linear arc-flow mathematical model and proposed a linearization method to reformulate the model into MILP models. More recently, Jetlund and Karimi (2003) considered a maximum-profit scheduling problem for a fleet of multi-compartment chemical tankers. Their MILP model with variable-length slots is capable of selecting routes and cargos for multiple ships optimally. A novel heuristic algorithm was proposed for solving large size problem by repeatedly solving the base formulation for every ship in the fleet.

2.3 Container Transshipment at Container Terminal

Container transshipment is a research area that extends from maritime shipping to other types of transportation modes. At a container terminal, the containers shipped by large vessels are transshipped to barges, trucks, trains, and so on. Because of the types of cargos (containers) and the different transportation modes (trucks, etc.) involved, the operation required is very different from the chemical transshipment operation we considered in this research work, where direct ship to ship transfers of chemical cargos are required.

The research in this area focuses on the detailed operations at the container terminal. For example, Vis and Koster (2003) considered the allocation of berths to ships, types of material handing equipment used for unloading and loading containers, more often, the crane scheduling problem. In addition, inter-terminal transfers and stacking of containers are also investigated.

2.4 Tanker Lightering

As seen from the above survey of literature in chemical shipping field, there is a complete absence of research on chemical transshipment except one special case, namely the crude or tanker lightering. Tanker lightering is the transfer of some crude oil from a large tanker to one or more small vessels in order to lighter the large tanker. It enables the tanker to then enter a shallow port or pass through shallow channels to a refinery's discharging docks, where it could not enter previously due to draft limitations. This is also a transshipment process, as cargos transfer directly from one carrier to another. Daskin and Walton (1982) described a general lightering problem for supertankers. They proposed a linked set of queuing models comprising a cyclic queuing model for lightering vessel operations and an approximate queuing model for tanker delays. The model is approximate and limited with several simplifying assumptions, for example, only one lightering vessel can service a large tanker at a time, and so on.

Later, Andrews et al. (1996) addressed a real-life crude oil lightering problem faced by a lightering service provider company in Delaware Bay. The company operates a fleet of lightering vessels to provide services for different refineries clustered in the region. The authors built a model employing some heuristic loading policies to simulate the lightering operations. The lightering company uses this model to study the effects of various policies on service level and profit, as well as the role of lightering operation in customer's crude oil supply chain.

Chajakis (1997) highlighted and stressed the importance and application of this model in crude oil transportation. He pointed out the fact that the advances of modeling in crude oil supply chain has helped the petrochemical companies to reduce significant logistics costs. The author developed a discrete-time MILP formulation together with some heuristics that incorporates the knowledge of an experienced fleet scheduler. His simulation model is also able to generate a short-term fleet lightering schedule within minutes even during congestion time. In a later work, Chajakis (2000) again addressed the importance of tanker lightering modeling in petroleum logistics. At the strategic planning level, a simulation model was built to assist decision making of optimal fleet size and composition. At the operational level, an optimization-based scheduling model was developed to generate good schedules of vessel assignments, service timings and lightering volumes. It is especially important during congestion time for the company and the anchorage.

In a recent work, Lin et al. (2003) studied the lightering fleet scheduling problem that may involve multiple voyages for small vessels to lighter the required amount of crude oil. They aimed to generate a short-term lightering schedule that minimizes the demurrage cost of tankers and the voyage cost of lightering vessels for the lightering service company. They proposed a continuous-time MILP formulation based on event points. Their model accommodated some practical lightering policies such as two-stage lightering (some very large tankers require lightering more than once) and a lightering vessel loading crude oil from multiple tankers during one voyage. However, their model has some serious limitations. First, they assumed single-compartment (single-parcel or single-crude) lightering vessels. They did not restrict the number of simultaneous services for a single tanker, while no more than two simultaneous services are possible in practice. They also did not allow a lightering vessel to lighter more than two tankers within one voyage. Furthermore, they ignored the differences in crude densities, travel times between tankers, and fuel consumption cost of lightering vessels, and did not allow the freedom to select the crudes to lighter. Often, these features are real and important in the tanker lightering problem.

2.5 Scope of Research

As seen from the above literature review, there is no previous work done in the field of scheduling general chemical transshipment operations. Yet, there is an increasing demand for transshipments in many industrial operations. A high quality schedule is of utmost importance as it can save substantial operating cost for the companies. In addition, there are several limitations of the existing optimization-based modeling for the tanker lightering problem. Therefore in this work, we focus our research in these two main areas.

In the first part, we develop three types of MILP models to address the scheduling of general chemical transshipment operations all using both slot-based and sequence-based approaches. In the models, nonlinear constraints are required. We then propose three alternate linearization methods. The models vary in numbers of binary/continuous variables, CPU times, and model tightness (relaxed mixed integer programming (rMIP) objective value). Therefore, we conduct a thorough study by testing them over several examples, and to investigate the factors affecting the model performance.

The second part of the research aims to develop a more realistic and practical optimization model which includes all the special and realistic features of tanker lightering operation. Our MILP model uses a mix of slot-based and pair-wise sequencing methods; whereas, the existing model uses an event-based formulation method. Hence, we also compare the performances of the two models. In addition, we develop two alternate linearization methods, and address two different objectives. One objective considers the time-charter cost of tankers, while the other considers the demurrage cost of tankers.

CHAPTER 3

SCHEDULING OF GENERAL CHEMICAL TRANSSHIPMENT OPERATIONS

In this chapter, we address the scheduling of general chemical transshipment operations involving direct ship (deep-sea carrier) to ship (short-sea carrier) transfer of liquid chemical. The industrial decisions include the timing of each cargo transshipment service and the cargo transshipment sequence in each carrier. We develop several MILP models addressing all the above decisions such that the total system cost of the entire operation is minimized. In what follows, we first describe the research problem in detail, demonstrate the motivation for this work using an example, then we formulate the problem with nine alternate (of three types) MILP models, as well as some additional constraints that improve the model performance.

3.1 Problem Description

To enable the carriage of multiple chemicals, most ships that transport chemicals in bulk amounts possess several (5-15) compartments to keep their chemical cargos separate. Figure 3.1 shows a picture of such a multi-compartment carrier. Typically, it has two types of compartments. Some compartments are accessible from only one side of the carrier, while others are accessible from both sides. We call the former one-sided compartments, while the latter two-sided compartments. The maritime transport industry refers to the two sides of a carrier as larboard (left or port side) and starboard (right side). If one stands facing towards a carrier's front (bow or stem vs. stern), then larboard is the left side and starboard is the right side. Therefore, as shown in Figure 3.1, we classify the cargos in a multi-compartment ship into three types: larboard, starboard, and two-sided. The first two types of cargos reside in one-sided compartments of a multi-compartment carrier and we call them one-sided cargos, while the two-sided cargos reside in two-sided compartments.

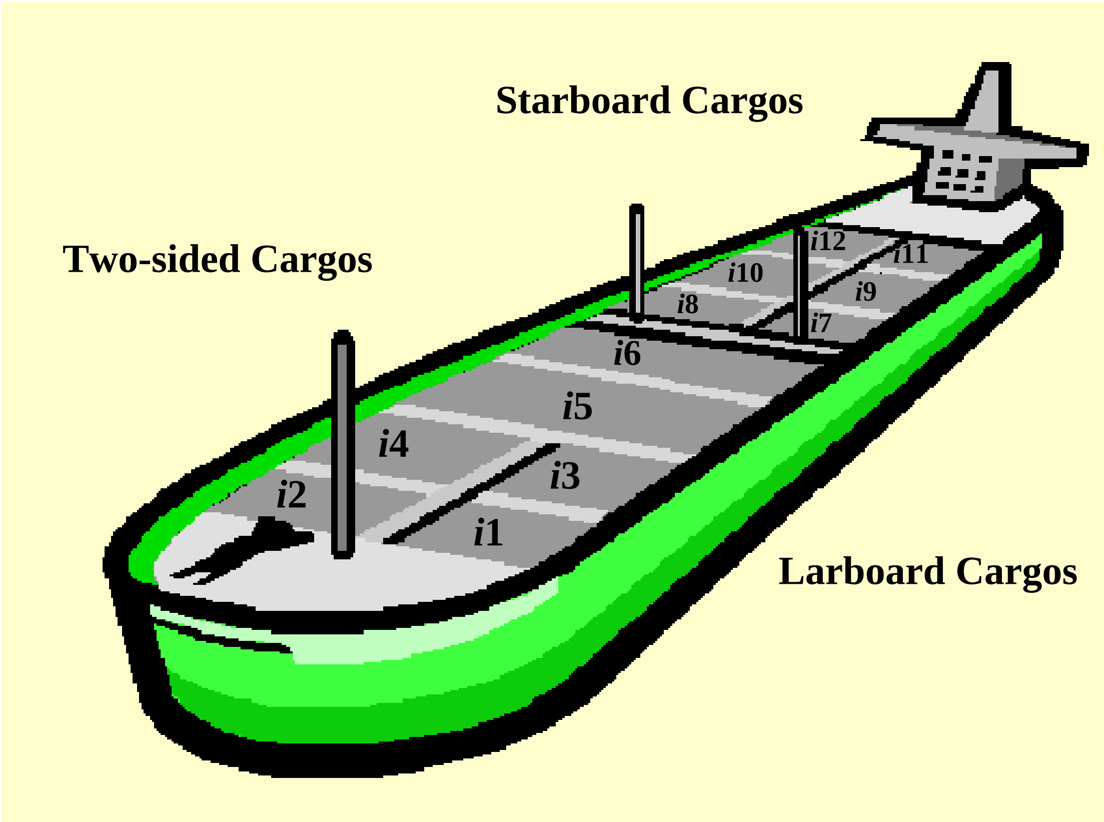

Figure 3.1: Schematic of a multi-compartment carrier

We consider a transshipment scenario (Figure 3.2) in which M deep-sea (or big) multi-compartment carriers or ships (m = 1, 2, ..., M) carrying several liquid chemicals anchor at a transshipment point (a seaport or seawater) at some known times. N short-sea (or small) carriers (n = 1, 2, ..., N) also reach the transshipment point at some known times to receive I cargos (i = 1, 2, ..., I) of various chemicals from the big ships for further regional transport. A cargo i is a lot of a single chemical transferred from one or more compartments that (1) belong to the same big ship, (2) carry the same chemical, (3) are of the same type (larboard, starboard, or two-sided), and (4) deliver to the same small ship. In other words, we merge the amounts of the same chemical coming from one or more same-sided compartments of a big ship and designated for the same small ship into one distinct transshipment cargo. The same chemical even though coming from different big ships also will go into different cargos. All transfers are direct, i.e. ship-to-ship from big to small ships. We divide the transshipment cargos into two sets, namely ***TSC*** = {*i* | cargo *i* is two-sided} and ***OSC*** = {*i* | cargo *i* is one-sided}. Furthermore, we define $ETAS_n$ ($ETAB_m$) as the time at which small (big) ship *n* (*m*) becomes ready to receive (transfer) its first cargo. Note that the big ships and small ships need not be deep-sea and short-sea carriers respectively. This work applies easily to any general scenario in which a set of ships transships a set of cargos to another set of ships.

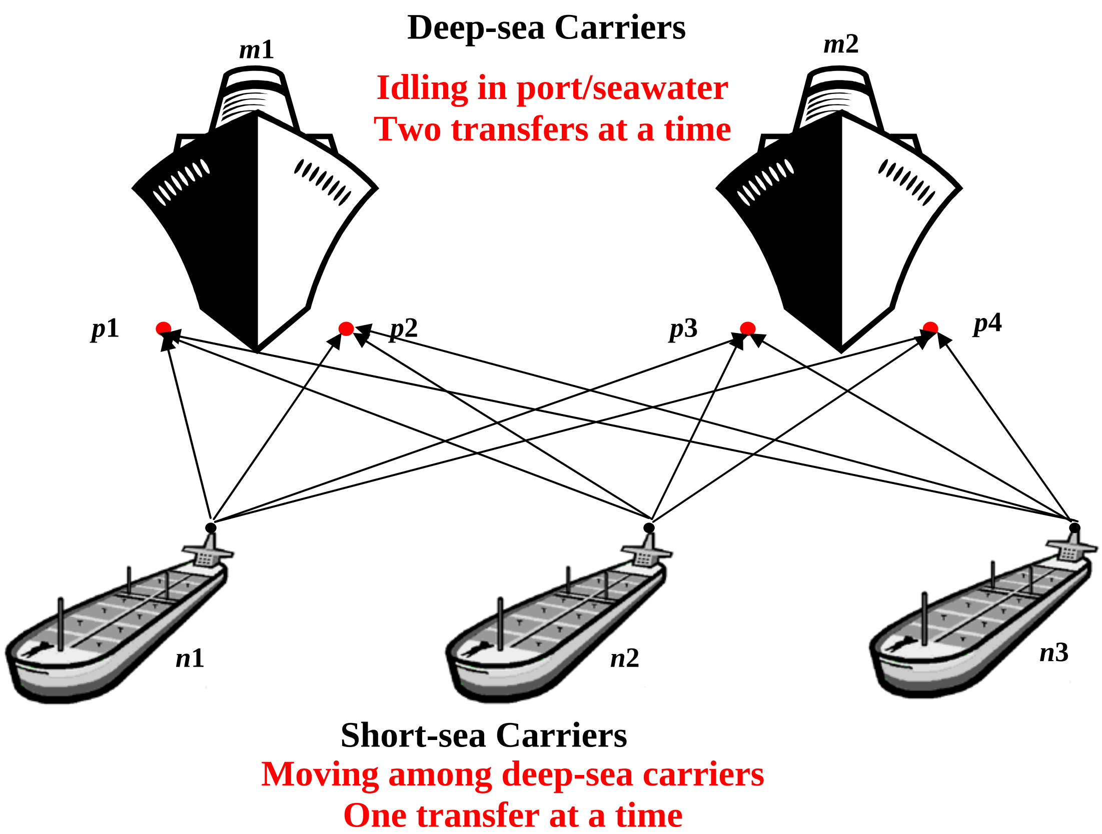

Figure 3.2: Chemical transshipment operations at transshipment location

Given the above, we wish to determine the sequence in, the sides (larboard or starboard) from, and the times at which, each small ship should receive cargos to minimize the total time-charter cost of all ships. The time-charter cost for a typical multi-parcel tanker can be in several tens of thousands of US$ per day. We compute it by knowing the total time that a ship spends at the transshipment point. In this problem, we assume the following:

(1) Each cargo, its amount, its origin (big) ship, its destination (small) ship, and its transfer rate are all fixed and known a priori.

(2) The cargo transfers can occur in any sequence. In practice, some restrictions may exist on the order in which a big ship may be able to transfer its cargos.

(3) The anchoring points (berths) of all big ships at the transshipment position (port) are fixed and known a priori. Furthermore, their number is not limiting and other restrictions related to the port's infrastructure are absent. Alternately, the anchoring times of the big ships are arranged a priori to avoid conflicts of berths. Often, a port may have limited berths and only a limited number of big ships can anchor at any time.

(4) Once a big ship anchors at a berth, it must finish transferring all its transshipment cargos, before it can move and leave. It cannot leave in the middle and anchor again. In other words, it is possible to transfer all its transshipment cargos from that berth. Thus, the big ships do not move once they anchor at berths, while the small ships do not anchor, but move among the big ships collecting their respective cargos. A more complex situation would involve a big ship moving and anchoring at different berths for different cargos.

(5) The fuel consumption of small ships during transshipment is negligible, so the distance that a small ship may need to move for collecting its cargos is

inconsequential.

(6) A big ship can transfer two cargos from different sides at the same time, while a small ship can receive only one cargo at a time.

3.2 Motivation

Singapore is a major hub for maritime chemical logistics due to its strategic position, deep waters, excellent port facilities, and infrastructure. It is the largest transshipment hub in Asia, where a significant portion of the 17,576 tankers visited Singapore for transshipment services during 2004 (MPA, 2004). The motivation for this work arises from our interaction with a major multi-national chemical shipping company in Asia-Pacific. This company has a large fleet of deep-sea intercontinental carriers that routinely visit Singapore to deliver and/or tranship various cargos for regional distribution. By studying the real transshipment operations of this company, we realized that manual, heuristic, and even ad hoc methods were used in practice to schedule these operations, which could easily mean substantial losses in efficiency in such a combinatorial problem. The application of systematic, discrete optimization methods could increase the efficiency of transshipment operations and result in substantial savings.

To get an estimate of potential savings, we select one example, namely Example 3a, from the examples that we use later in Chapter 4 (Example data in Chapter 4). This example involves two big ships ($M = 2$) transshipping sixteen cargos ($I = 16$) to four small ships ($N = 4$). We can use some common-sense policies or heuristics described below to derive a schedule (Figure 3.3) manually.

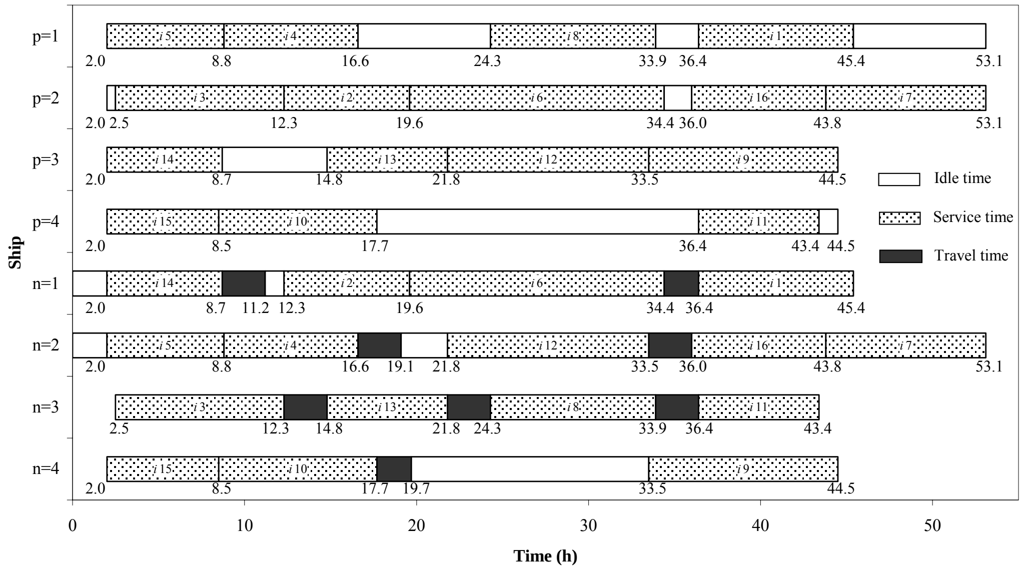

Figure 3.3: Gantt chart for the manual transshipment schedule of Example 3a based on common-sense policies / heuristics

Because our objective is to minimize the total time-charter cost of all carriers, a good schedule would minimize the time that the carriers spend idling or waiting for transshipment service. Now, the big ships are relatively more expensive than the small ones, so it would make sense to minimize their waiting times first. To this end, we employ a mix of a first-come first-serve and a priority-based approach. We divide all cargos into two groups. One that is "available" for service at any time; and the other is not. When a big ship arrives at the transfer location, we consider as available all of its transshipment cargos whose recipient ships are ready to receive. The remaining cargos become available subsequently, as and when their recipient ships arrive at the transfer location or finish previous services. In general, multiple cargos will be available at any time, so we assign priorities to cargos based on the time-charter costs of ships, cargo service times, etc. Thus, the set of available cargos will vary with time and cargo service status. If a cargo is engaged in a transshipment service, we treat the other cargos at the same position as unavailable for transfer at that time. They become available only after the current cargo finishes its transfer. Similarly, we also treat other cargos for the same recipient ship as unavailable, until this recipient ship finishes servicing the current cargo. Finally, as time progresses and activities start/finish, cargos become available at arbitrary times, so we repeatedly update our list of available cargos dynamically. Thus, whenever a change occurs in available cargos, we redo the ranking of cargos before we further develop the transshipment sequence.

To rank the available cargos, we use the following thumb rules and priorities in the order stated below:

1. A cargo whose big ship has the highest time-charter cost has the higher priority.
2. A cargo whose small ship has the highest time-charter cost is the next in priority.
3. A cargo with a shorter service time is the next in priority.
4. Serve consecutively the cargos that have the same unloading position and small ship.
5. For a two-sided cargo, select the transfer position such that the cargos on both sides of the big ship are evenly balanced.

Using the above priorities and heuristics, we determine a sequence in which each position of each big ship will transfer its cargos.

For this example, the estimated ready times of the big ships are 53.1 h and 44.5 h and those for the small ships are 45.4 h, 53.1 h, 43.4 h, and 44.5 h respectively. The total time-charter cost of the manual schedule is $31715.50. If we were to use a model such as the one presented in this chapter, we can obtain an optimal schedule (Figure 3.4) with a time-charter cost of $29829.00. This schedule represents a savings of 6.32%. Thus, the savings are substantial even for this simple example. Many practical scenarios can be much more complex with much greater savings, as it is hard to manually enumerate all possible combinations to generate good schedules. Of course, there would be no guarantee of an optimal schedule using any ad hoc procedure.

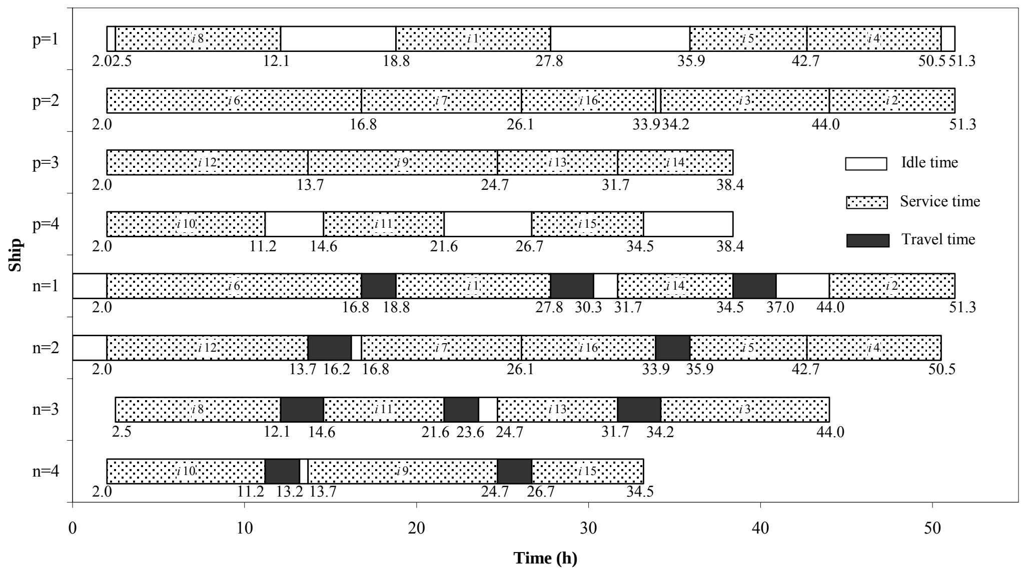

Figure 3.4: Gantt chart for the optimal transshipment schedule of Example 3a

3.3 MILP Formulations

A big ship can transfer a two-sided cargo ($i \in \boldsymbol{TSC}$) from either larboard or starboard. Therefore, the optimizer must decide the side from which the transfer should occur. To facilitate this decision, we replace each two-sided cargo by two one-sided cargos of the same size, namely a larboard cargo and a starboard cargo. Then, we will allow only one of these two cargos to transfer. Of course, this increases the number of cargos in our problem from I to $J = I + |\boldsymbol{TSC}|$, where $|\boldsymbol{TSC}|$ denotes the cardinality of $\boldsymbol{TSC}$. However, dealing with only one-sided cargos makes modeling easier. We use j to refer to these J redefined one-sided cargos, thus we now have j = 1, 2, ..., J one-sided transshipment cargos. Let set $\boldsymbol{J} = \{j$ | one-sided cargo derived from the original I transshipment cargos$\}$.

Now, each big ship has two sides for cargo transfer, so M ships have $2M$ possible ship-side combinations or positions (p = 1, 2, ..., $2M$) from which a cargo may transfer. Therefore, we define $\boldsymbol{P}_m$ = {p | position p belongs to big ship m} and divide $\boldsymbol{J}$ into $2M$ mutually exclusive and exhaustive subsets defined by $\boldsymbol{J}_p$ = {j | cargo j must transfer from position p}, $p = 1, 2, ..., 2M$. Furthermore, we define $\boldsymbol{JS}_n$ = {j | cargo j is destined for small ship n}, $\boldsymbol{IS}_n$ = {i | small ship n is to receive transshipment cargo i}, and $\boldsymbol{JB}_m$ = {j | cargo j is on big ship m}.

As stated earlier in the problem statement, each cargo j resides in a unique big ship and is destined for a unique small ship. Since we know its volume and associated ships (both big and small), we can estimate the time τ_j (service time) required to transfer cargo j. Note that τ_j represents the total time required for serving cargo j including anticipated delays, hose-connection time, actual transfer time, hose-disconnection time, etc. In addition to the service time, a small ship will need time to travel from one position to another. Since we know the anchoring positions of all big ships, we can estimate the time $\theta_{npp'}$ required by small ship n to travel from position p to p'.

Lastly, we define a unique time $ETAC_j$ at which cargo j becomes available for transshipment based on the earliest times at which its big ship can transfer and small ship can receive. Clearly, $ETAC_j = \max[ETAS_n, ETAB_m]$ with n and m such that $j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m$.

We now present our three alternate continuous-time formulations, namely F1, F2, and F3. Of these, F2 and F3 are largely similar to each other. In F1 (F2, F3), we first model the operation of each small (big) ship, followed by that of each big (small) ship, and then we couple both operations appropriately.

3.3.1 Formulation F1

3.3.1.1 Short-Sea Carriers

We model time on a small ship n in terms of a series of K_n chronologically ordered contiguous slots (k = 1, 2, ..., K_n) of variable lengths. Then, we assign exactly one cargo to each slot. Figure 3.5 shows a schematic for this slot-based modeling approach. Since we know the cargos that small ship n is to receive, $K_n = |\boldsymbol{IS}_n|$, i.e., the number of slots equals the number of real transshipment cargos exactly. A new slot begins, whenever ship n starts servicing a new cargo. During each slot, a small ship n typically performs three tasks. First, it receives a cargo from a big ship. Then, it travels from its current position to the position of its next cargo. This may mean moving to another big ship and even changing sides. Lastly, it waits to begin receiving the next cargo. Note that a ship n may wait for some time after reaching near the next position, as another small ship may be at the next position. It will move into the receiving position, when the position is free. In what follows, we write each constraint for all valid values of its defining indices, unless stated otherwise.

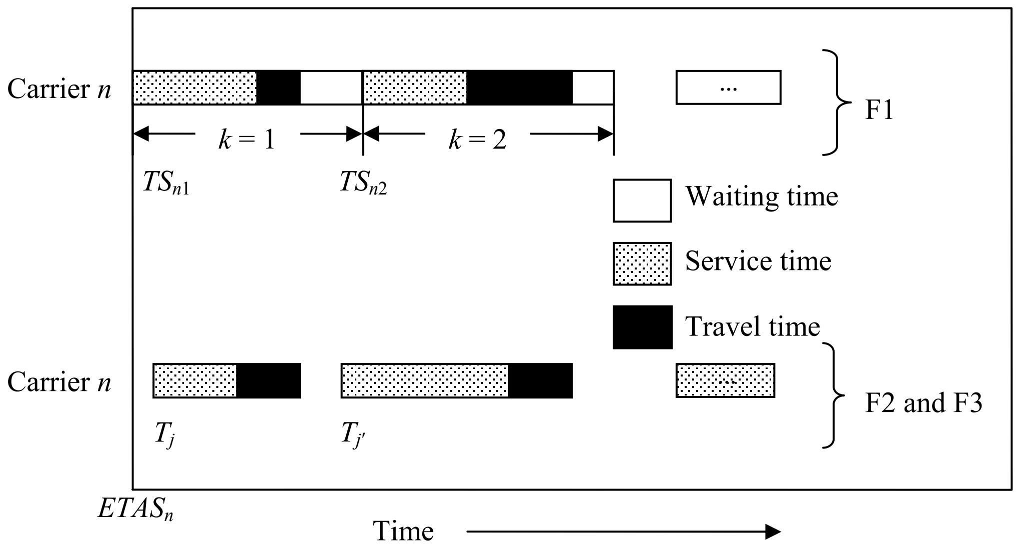

Figure 3.5: Slot-based/sequence-based approaches for the short-sea carriers

We must first assign a unique cargo to each slot of ship n. Because ship n has K_n slots, a cargo for ship n can go into one of K_n slots. Thus, we imagine that a cargo j has $K_j = K_n$ target or destination slots with $n \ni j \in \boldsymbol{JS}_n$. Now, to assign cargo j to one of these slots, we define a binary variable x_{jk} as:

$$x_{jk} = \begin{cases} 1 & \text{if cargo } j \text{ transfers during slot } k \\ 0 & \text{otherwise} \end{cases} \qquad 1 \le k \le K_j$$

To ensure exactly one cargo in each slot of ship n, we write the following for every n,

$$\sum_j x_{jk} = 1 \qquad j \in \boldsymbol{JS}_n \qquad (3.1)$$

Recall that we represent each cargo $i \in \boldsymbol{TSC}$ by two one-sided cargos, only one of which will transfer. As a result, a cargo j may not transfer at all. Therefore, to monitor the service status of a cargo j, we define a 0-1 continuous variable U_j as:

$$U_j = \begin{cases} 1 & \text{if cargo } j \text{ transfers} \\ 0 & \text{otherwise} \end{cases} = \sum_{k=1}^{K_j} x_{jk}$$

Clearly, $U_j = 1$, if $i(j) \in \boldsymbol{OSC}$, where $i(j)$ is the parent, real cargo i of j. Throughout this problem, we replace U_j by 1 for $j \ni i(j) \in \boldsymbol{OSC}$. On the other hand, $U_{j1} + U_{j2} = 1$, if $i(j1) = i(j2) \in \boldsymbol{TSC}$ and $j1$ and $j2$ respectively are the starboard and larboard cargos representing a cargo $i \in \boldsymbol{TSC}$. Therefore, we get,

$$\sum_{k=1}^{K_j} x_{jk} = 1 \qquad j \ni i(j) \in \boldsymbol{OSC} \qquad (3.2a)$$

$$\sum_{k=1}^{K_j} x_{jk} = U_j \qquad j \ni i(j) \in \boldsymbol{TSC} \qquad (3.2b)$$

$$U_{j1} + U_{j2} = 1 \qquad j1 \;\&\; j2 \ni i(j1) = i(j2) \in \boldsymbol{TSC} \qquad (3.2c)$$

Let TS_{nk} be the start time of slot k on small ship n. Because ship n can receive a cargo only after its earliest cargo becomes available, we have $TS_{nk} \ge MTS_n = \min_j(ETAC_j)$ with $j \in \boldsymbol{JS}_n$. Furthermore, eq. 3.1 ensures that each slot k will have exactly one cargo, so TS_{nk} must be after the time at which the assigned cargo becomes available. In other words,

$$TS_{nk} \geq \sum_j ETAC_j x_{jk} \qquad j \in \boldsymbol{JS}_n \qquad (3.3)$$

From Figure 3.5, the length $[TS_{n(k+1)}–TS_{nk}]$ for slot k on ship n should include the times for service, wait, and travel from one transfer position to another. It is easy to compute the service time for a slot k, but not the travel time. This is because we need to know the transfer positions between which a small ship moves. To this end, we first identify the current position of a small ship by using the following binary variable, which we will later eliminate from our formulation.

$$z_{npk} = \begin{cases} 1 & \text{if small ship } n \text{ is at position } p \text{ during slot } k \\ 0 & \text{otherwise} \end{cases} = \sum_j x_{jk} \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{J}_p$$

This variable helps us model the transitions of ship n from one position to another by defining another 0-1 continuous variable as follows:

$$Z_{npp'k} = \begin{cases} 1 & \text{if small ship } n \text{ moves from position } p \text{ to } p' \text{ during slot } k \\ 0 & \text{otherwise} \end{cases}$$

$$p \;\&\; p' \in \boldsymbol{P}_n,\ k < K_n$$

where, $\boldsymbol{P}_n = \{p \mid \text{small ship n receives a cargo from position } p\}$. We can express the variable as $Z_{npp'k} = z_{npk} z_{np'(k+1)}$, which we linearize exactly by using,

$$\sum_{p'} Z_{npp'k} = z_{npk} = \sum_j x_{jk} \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{J}_p,\ p \;\&\; p' \in \boldsymbol{P}_n,\ k < K_n \qquad (3.4a)$$

$$\sum_{p'} Z_{np'pk} = z_{np(k+1)} = \sum_j x_{j(k+1)} \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{J}_p,\ p \;\&\; p' \in \boldsymbol{P}_n,\ k < K_n \qquad (3.4b)$$

Equations (3.4a,b) force $Z_{npp'k}$ to be binary automatically and we no longer need z_{npk} in our formulation. In addition, if a small ship n does not receive any cargo from either p or p', then $Z_{npp'k} = 0$. Hence, we fix $Z_{npp'k} = 0$ for $p \notin \boldsymbol{P}_n$ or $p' \notin \boldsymbol{P}_n$, and $k < K_n$. Also, if

$|\boldsymbol{JS}_n \cap \boldsymbol{J}_p| = 0$, then we do not need eqs. 4a,b. With these, we write the constraint for slot length as:

$$TS_{n(k+1)} \geq TS_{nk} + \sum_j \tau_j x_{jk} + \sum_p \sum_{p'} \theta_{npp'} Z_{npp'k} \qquad j \in \boldsymbol{JS}_n,\, p \,\&\, p' \in \boldsymbol{P}_n,\, k < K_n \qquad (3.5)$$

where, the second term on the right side is the service time of cargo j in slot k of ship n, and the third represents the travel time.

3.3.1.2 Deep-Sea Carriers

In contrast to the small ships, we now use a pair-wise sequencing approach (Figure 3.6) for modeling the sequence in which a big ship transfers its cargos. This involves defining a slot for each cargo, whose length depends on its transfer status. If the big ship does not transfer that cargo, then the slot length is zero, otherwise it equals the transfer time. To model the cargo sequence, we define a binary variable that sequences two cargos relative to each other at a position:

$$y_{jj'} = \begin{cases} 1 & \text{if cargo } j' \text{ transfes later than cargo } j \\ 0 & \text{otherwise} \end{cases} \qquad p(j) = p(j'),\, n(j) \neq n(j'),\, j < j'$$

where, $n(j)$ is $n \ni j \in \boldsymbol{JS}_n$ and $p(j)$ is $p \ni j \in \boldsymbol{J}_p$. Note that we define the above only for those cargo pairs that transfer from the same position p, but are destined for different small ships. We exclude the cargo pairs that belong to the same small ship, as the earlier slot-based sequencing of the small ships already ensures their relative order. Moreover, we define the above binary variable only for the combinations of j and j', and not for permutations.

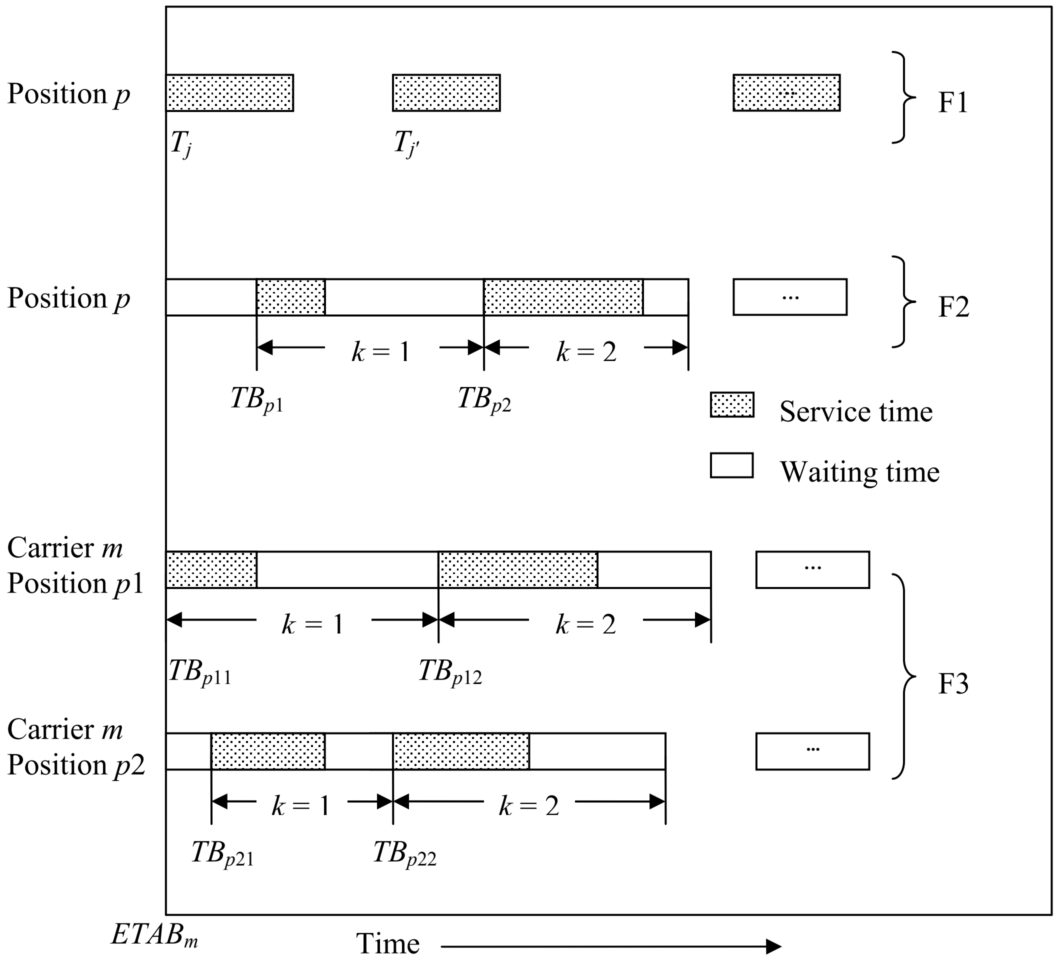

Figure 3.6: Slot-based/sequence-based approaches for the deep-sea carriers

Because a big ship cannot transfer multiple cargos at the same time from any given position, we must spread their service times sufficiently to eliminate clashes. The slot-based sequencing ensured this for the cargos belonging to the same small ship, but we need additional constraints for the cargos belonging to different small ships. To this end, we define T_j as the time at which cargo j transfers. If another cargo j' with $p(j) = p(j')$ and $n(j) \neq n(j')$ transfers later than j, then the following must hold,

$$T_{j'} \geq T_j + \tau_j U_j - H\left(1 - y_{jj'}\right) \qquad p(j) = p(j'),\ n(j) \neq n(j'),\ j < j' \qquad (3.6a)$$

$$T_j \geq T_{j'} + \tau_{j'} U_{j'} - H y_{jj'} \qquad p(j) = p(j'),\ n(j) \neq n(j'),\ j < j' \qquad (3.6b)$$

where, H is a big-M constant greater than the horizon time at the position $p(j) = p(j')$. The second terms on the right sides are the service times of cargos j and j', which are zero if they do not transfer. If a cargo does not transfer, then its slot length will be zero. Note that $U_j = 1$, if $i(j) \in \boldsymbol{OSC}$. Equation 3.6a (3.6b) relaxes, when $y_{jj'} = 0$ ($y_{jj'} = 1$). Both equations (3.6a,b) are necessary, because we defined $y_{jj'}$ for $j < j'$ only and we must consider all permutations of cargos.

Clearly, a cargo cannot start its transfer, until it becomes available. Therefore, if a cargo j transfers ($U_j = 1$), then the earliest time that it can transfer is $ETAC_j$. However, a cargo j with $i(j) \in \boldsymbol{TSC}$ may not transfer at all, if $U_j = 0$. Then, what should be the transfer time of such a cargo? If we let such a discarded cargo float in the middle of a cargo sequence, then it would lead to an erroneous travel time between its immediate cargo neighbors in the sequence. To avoid this error, we force each cargo with $U_j = 0$ to be before all transferred cargos and to start at the earliest possible start time for any cargo at its respective position, which is $MT_j = \min_{j'}(ETAC_{j'})$ with $j' \ni p(j') = p(j)$. We can achieve these two requirements by using,

$$T_j \geq ETAC_j U_j + MT_j \left(1 - U_j\right) \tag{3.7a}$$

$$T_j \leq MT_j + HU_j \tag{3.7b}$$

Eqs. 3.7a,b force $T_j = MT_j$, when $U_j = 0$ and $T_j \geq ETAC_j$, when $U_j = 1$. We set the lower bound of T_j as $ETAC_j$ for $j \ni i(j) \in \boldsymbol{OSC}$ and MT_j for $j \ni i(j) \in \boldsymbol{TSC}$.

So far, we ensured the sequencing and timings on each ship (big & small) separately, but the key feature that differentiates chemical transshipment (ship-to-ship in general) from non-chemical transshipment (ship-to-ship not required in general) is that both big and small ships must have coupled and synchronized schedules. The service intervals for each cargo on both ships must match exactly in time. Therefore, we need some coupling constraints.

3.3.1.3 Coupling of Ship Operations

To ensure that both big and small ships engage with any given cargo simultaneously, we must demand that the transfers begin at the same time on both ships for every cargo. Since T_j is the time at which a cargo j begins transfer, it must equal the time at which its small ship begins to receive it. In other words,

$$T_j = \sum_{k=1}^{K_j} TS_{nk} x_{jk} \qquad n = n(j)$$

$$TS_{nk} = \sum_{j} T_j x_{jk} \qquad j \in \boldsymbol{JS}_n \qquad (3.8a)$$

However, the first equation implies $T_j = 0$, when $U_j = 0$. As discussed earlier, T_j should be at its lower bound MT_j, when $U_j = 0$. Therefore, we modify the first equation as:

$$T_j = \sum_{k=1}^{K_j} TS_{nk} x_{jk} + MT_j\left(1 - U_j\right) \qquad n = n(j) \qquad (3.8b)$$

Eqs. 3.8a,b are nonlinear constraints and we can replace them by the following two linear constraints with the help of eq. 3.7b.

$$T_j \geq TS_{nk} - H(1 - x_{jk}) \qquad n = n(j) \qquad (3.9a)$$

$$T_j \leq TS_{nk} + H(1 - x_{jk}) \qquad n = n(j) \qquad (3.9b)$$

Alternately, we can use either eq. 3.9a or eqs. 3.7b and 3.9b along with the following equality that we can derive from eq. 3.8b by summing both sides over $j \in \boldsymbol{JS}_n$ for every n,

$$\sum_j T_j = \sum_{k=1}^{K_n} TS_{nk} + \sum_j MT_j\left(1 - U_j\right) \qquad j \in \boldsymbol{JS}_n \qquad (3.10)$$

Note that $K_j = K_n$ for $j \in \boldsymbol{JS}_n$. If we use eq. 3.10, then one big-M constraint, namely eq. 3.9b, and eq. 3.7b become redundant; eq. 3.9a suffices. Thus, the use of eq. 3.10 eliminates some big-M and other constraints. Therefore, we prefer eqs. 3.10 and 3.9a over eqs. 3.7b, 3.9a, and 3.9b.

One more alternative exists for linearizing eq. 3.8b. This involves defining a continuous variable $TX_{jk} = TS_{nk}x_{jk} = T_jx_{jk}$ with $n = n(j)$. Substituting this into eqs. 3.8a,b gives us,

$$TS_{nk} = \sum_{j} TX_{jk} \qquad j \in \boldsymbol{JS}_n \qquad (3.11a)$$

$$T_j = \sum_{k=1}^{K_j} TX_{jk} + MT_j\left(1 - U_j\right) \qquad (3.11b)$$

Lastly, we force $TX_{jk} = 0$, when $x_{jk} = 0$.

$$TX_{jk} \le Hx_{jk} \qquad (3.11c)$$

The linearization comprising eqs. 3.11a-c requires more continuous variables than the first two alternatives, but fewer constraints and nonzeros than the first alternative (eqs. 3.7b and 3.9a-b). Theoretically, it may be tighter, because it has no big-M constraint. However, the increase in the number of continuous variables may slow it sufficiently to outweigh the advantage from tightness. Thus, it is not clear if F13 can be the fastest-solving formulation. Of the three alternative linearizations, the second one (eqs. 3.9a and 3.10) seems the best intuitively, but hardcore computational performance is the only way to decide among these three alternatives.

<u>3.3.1.4 Scheduling Objective</u>

As stated earlier, the scheduler's goal is to minimize the total time-charter cost of all carriers. To this end, we must determine the departure time of each ship, for which we have two ways.

Let DTB_m be the time at which big ship m ends its operations. Clearly, it cannot leave, until it transfers all its cargos, so

$$DTB_m \geq T_j + \tau_j U_j \qquad j \in \boldsymbol{JB}_m \qquad (3.12a)$$

We obtained the above by considering the time-axis of a big ship. Alternately, we can also consider the time-axis of a small ship. Since each operation starts simultaneously on both big and small ships, we can also compute DTB_m from the end times of slots on the small ships, provided m does serve a cargo in that slot. In other words, we have,

$$DTB_m \geq TS_{nk} + \sum_j \tau_j x_{jk} - H(1 - \sum_j x_{jk}) \qquad j \in \boldsymbol{JB}_m \cap \boldsymbol{JS}_n \qquad (3.12b)$$

Note that eq. 3.12b relaxes, when big ship m is not involved in slot k. It is not clear, if either eq. 3.12a or 3.12b is redundant, so we keep both.

The slot-based sequencing makes it easy to obtain the departure time DTS_n of a small ship n. It is when the ship's last slot ends. In addition, a ship cannot depart earlier than its arrival time. We impose lower bounds, $DTB_m \geq \min_j(MT_j)$ with $j \in \boldsymbol{JB}_m$ and $DTS_n \geq MTS_n$.

With this, the total time-charter cost of all ships ($TTCC$) is,

$$TTCC = \sum_m TCCB_m \left(DTB_m - ETAB_m\right) + \sum_n TCCS_n \left(TS_{nK_n} + \sum_{j \in \boldsymbol{JS}_n} \tau_j x_{jK_j} - ETAS_n \right) \qquad (3.13)$$

where, $TCCB_m$ and $TCCS_n$ are the time-charter costs of big ship m and small ship n respectively.

This completes our three alternate versions (F11, F12, and F13) of F1. Eqs. 3.1-3.7a, 3.12, and 3.13 are common to all three. The rest are eqs. 3.7b and 3.9a,b for F11, eqs. 3.9a and 3.10 for F12, and eqs. 3.11a-c for F13.

3.3.2 Formulation F2

In F1, we used a slot-based sequencing approach for the small ships and a pair-wise sequencing approach for the big ships. In this section, we switch the modeling approaches, i.e. we use the slot-based (pair-wise) sequencing for big (small) ships. However, the underlying concepts in F2 are the same as those in F1, so we merely identify the main differences and state the modified constraints. In F2, each transfer position of a big ship is analogous to a small ship in F1, and each small ship is analogous to a position in F1. The main differences are that the positions do not move in F2 in contrast to the small ships in F1, so transitions are easier to model. Furthermore, the numbers of slots and binary variables are different, on which we elaborate later.

Using the above analogies, we partition the time horizon for each position p into $K_p = |\boldsymbol{J}_p|$ contiguous slots ($k = 1, 2, ..., K_p$) as shown in Figure 3.6. Then, we set $K_j = |\boldsymbol{J}_p|$ with $p = p(j)$ and define x_{jk} as in F1 and TB_{pk} as the time at which slot k begins at position p. Note that x_{jk} now refers to slots for a position rather than a small ship. Then, following the logic behind eq. 3.1, we write for every p,

$$\sum_j x_{jk} \leq 1 \qquad j \in \boldsymbol{J}_p \qquad (3.14)$$

And, eqs. 3.2a-c apply as is. Note that, unlike F1, if a two-sided cargo does not transfer from a given position, then we would have an empty slot, therefore eq. 3.14 is an inequality rather than an equality. Eq. 3.14 allows a slot to be without a cargo. We force such empty slots to start at the earliest possible start time at the given position, which we define as $MTB_p = \min_j(ETAC_j)$ with $j \ni \boldsymbol{J}_p$.

$$TB_{pk} \geq \sum_j ETAC_j x_{jk} + MTB_p \left(1 - \sum_j x_{jk}\right) \qquad j \in \boldsymbol{J}_p \qquad (3.15a)$$

$$TB_{pk} \le MTB_p + H\sum_j x_{jk} \qquad j \in \boldsymbol{J}_p \qquad (3.15b)$$

Note that when a slot has no cargo, its start time becomes MTB_p or the lower bound of TB_{pk}, i.e. the slot precedes all non-empty slots.

Because positions do not move during transitions, we do not need eqs. 3.4 in F2 and the slot length constraint (eq. 3.5) becomes,

$$TB_{p(k+1)} \ge TB_{pk} + \sum_j \tau_j x_{jk} \qquad j \in \boldsymbol{J}_p,\ k < K_p \qquad (3.16)$$

For the pair-wise sequencing on the small ships, we define:

$$y_{jj'} = \begin{cases} 1 & \text{if cargo } j' \text{ transfers later than cargo } j \\ 0 & \text{otherwise} \end{cases} \qquad n(j) = n(j'),\ p(j) \ne p(j'),\ j < j'$$

Note that we define the above variable for cargo pairs with the same small ship, but different positions.

In contrast to F1, the modeling of travel times does not need any transition variables, because $y_{jj'}$ provides this information readily (Figure 3.5). Thus, following eqs. 3.6a,b, we write,

$$T_{j'} \ge T_j + [\tau_j + \theta_{np(j)p(j')}]U_j - H\left(1 - y_{jj'}\right) \qquad n = n(j) = n(j'),\ p(j) \ne p(j'),\ j < j' \quad (3.17a)$$

$$T_j \ge T_{j'} + [\tau_{j'} + \theta_{np(j')p(j)}]U_{j'} - Hy_{jj'} \qquad n = n(j) = n(j'),\ p(j) \ne p(j'),\ j < j' \quad (3.17b)$$

Note that eqs. 3.17a,b do not count the transfer time from a cargo that does not transfer.

Eqs. 3.7a,b remain the same as in F1 except that $MT_j = \min_{j'}(ETAC_{j'})$ with $n(j') = n(j)$. The lower bounds of T_j remain unchanged from F1.

The methods for coupling the big and small ship operations are the same as in F1. The analogs of eqs. 3.8a,b are,

$$TB_{pk} = \sum_j T_j x_{jk} + MTB_p\left(1 - \sum_j x_{jk}\right) \qquad j \in \boldsymbol{J}_p \qquad (3.18a)$$

$$T_j = \sum_{k=1}^{K_j(K_p)} TB_{pk} x_{jk} + MT_j \left(1 - U_j\right) \qquad p = p(j) \qquad (3.18b)$$

Following the linearization methods in F1, we rewrite eqs. 3.9a,b, 3.10, and 3.11a as,

$$T_j \geq TB_{pk} - H\left(1 - x_{jk}\right) \qquad p = p(j) \qquad (3.19a)$$

$$T_j \leq TB_{pk} + H\left(1 - x_{jk}\right) \qquad p = p(j) \qquad (3.19b)$$

$$\sum_j T_j + \sum_{k=1}^{K_p} MTB_p \left(1 - \sum_j x_{jk}\right) = \sum_{k=1}^{K_p} TB_{pk} + \sum_j MT_j \left(1 - U_j\right) \qquad j \in \boldsymbol{J}_p \qquad (3.20)$$

$$TB_{pk} = \sum_j TX_{jk} + MTB_p \left(1 - \sum_j x_{jk}\right) \qquad j \in \boldsymbol{J}_p \qquad (3.21a)$$

where, $TX_{jk} = TB_{pk}x_{jk} = T_jx_{jk}$ with $p = p(j)$.

Then, for the departure times, the analogs of eqs. 3.12a,b are:

$$DTS_n \geq T_j + \tau_j U_j \qquad j \in \boldsymbol{JS}_n \qquad (3.22a)$$

$$DTS_n \geq TB_{pk} + \sum_j \tau_j x_{jk} - H(1 - \sum_j x_{jk}) \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{J}_p \qquad (3.22b)$$

For a big ship, the departure time will be the start time of the last slot plus the service time of the cargo in that slot. However, each big ship has two transfer positions, so the final departure time will be the maximum of either side, or,

$$DTB_m \geq TB_{pK_p} + \sum_j \tau_j x_{jK_p} \qquad p \in \boldsymbol{P}_m, j \in \boldsymbol{J}_p \qquad (3.23)$$

The lower bounds of DTB_m and DTS_n remain unchanged from F1.

Finally, we rewrite the objective function (eq. 3.13) as,

$$TTCC = \sum_m TCCB_m \left(DTB_m - ETAB_m\right) + \sum_n TCCS_n \left(DTS_n - ETAS_n\right) \qquad (3.24)$$

This completes our second formulation. Again, we have three alternatives here. F21 uses eqs. 3.15b, 3.7b, and 3.19a,b; F22 uses eqs. 3.15b, 3.7b, 3.19a, and 3.20; F23 uses eqs. 3.21a with 3.11b,c; and all three use eqs. 3.14, 3.15a, 3.16-3.17, and 3.22-3.24 in addition to the unchanged equations from F1.

3.3.3 Formulation F3

In formulation F2, we used slot-based (pair-wise) sequencing for big (small) ships and allowed at most one cargo per slot. In this formulation, we follow the same sequencing methods, but instead of using slots at each position, we use slots for each big ship. This would allow us to have two cargos per slot (one at each position of a big ship) and should require a different number of binary variables than F2. In the following, we identify only the main differences from F2.

We divide the time horizon for each big ship m into K_m contiguous slots (k = 1, 2, ..., K_m) as shown in Figure 6. Clearly, $K_m = \max_p |\boldsymbol{J}_p|$ for $p \in \boldsymbol{P}_m$. We set $K_j = K_m$ with $m \ni j \in \boldsymbol{JB}_m$, $K_p = K_m$ with $m \ni p \in \boldsymbol{P}_m$, and define x_{jk} as in F1. Though two cargos can reside in one slot, each position of a slot can transfer at most one cargo, so eq. 3.14 remains unchanged. Similarly, although we defined slots with respect to big ship rather than position, the variables TB_{pk}, T_j, TX_{jk}, DTS_n, and DTB_m and all the constraints and lower bounds remain the same as in F2.

Again, we have three alternate formulations as in F2 except for the proper values of K_j and K_p, and the numbers of continuous and binary variables.

In principle, we could formulate F3 in terms of real cargos (i) rather than derived cargos (j). On the face of it, that would seem to reduce the binary variables even further. However, a detailed analysis reveals that we still must make a choice of transfer position for each two-sided cargo and we would need a binary variable for that decision. In other words, it is just not possible to use a single binary variable for two-sided cargos and this makes fruitless the idea of formulating F3 in terms of real cargos.

Considering the fact that the only difference between F2 and F3 is the number of binary variables, it would be instructive to know which uses fewer. As we show later in section 6, F3 always uses more binary variables than F2 and thus we expect it to be inferior to F2. We confirm this conclusion later via detailed numerical evaluation.

3.4 Additional Constraints

Apart from the constraints stated above for F1, F2, and F3, we can write additional constraints that explicitly sequence empty cargos and slots. These constraints involve binary variables only and may yield tighter and more efficient formulations.

As discussed earlier, a cargo j with $i(j) \in \boldsymbol{TSC}$ may not transfer at all models. We forced such discarded cargo ($U_j = 0$) to be before each transferred cargo ($U_j = 1$) and to start at the earliest possible start times of ships or positions in the models. The following two constraints achieve this objective for F1,

$$y_{jj'} \geq U_{j'} - U_j \qquad p(j) = p(j'),\ n(j) \neq n(j'),\ i(j) \text{ or } i(j') \in \boldsymbol{TSC},\ j < j' \qquad (3.25a)$$

$$y_{jj'} \leq U_{j'} - U_j + 1 \qquad p(j) = p(j'),\ n(j) \neq n(j'),\ i(j) \text{ or } i(j') \in \boldsymbol{TSC},\ j < j' \qquad (3.25b)$$

Note that we write the above, only when at least one of j and j' is a one-sided cargo that models an original two-sided cargo. When a cargo j transfers ($U_j = 1$) and j' does not ($U_{j'} = 0$), then eq. 3.25a relaxes and eq. 3.25b forces $y_{jj'} = 0$. Thus, each non-transferred cargo j' precedes every transferred cargo j. When a cargo j does not transfer ($U_j = 0$) and j' transfers ($U_{j'} = 1$), then eq. 3.25a forces $y_{jj'} = 1$ and eq. 3.25b relaxes. Thus, each transferred cargo j' succeeds each non-transferred cargo j. If neither cargo j nor j' transfers (nor both transfer), then both eqs. 3.25a and 3.25b relax and impose no restriction on sequencing. Interestingly, we can combine eqs. 3.25a and 3.25b into one single equality constraint as follows:

$$y_{jj'} + \delta_{jj'} = U_{j'} - U_j + 1 \qquad p(j) = p(j'),\ n(j) \neq n(j'),\ i(j) \text{ or } i(j') \in \boldsymbol{TSC},\ j < j' \qquad (3.26)$$

where, $0 \leq \delta_{jj'} \leq 1$ is a continuous 0-1 variable.

The analogous constraint for F2 and F3 is the same as eq. 3.26 except for the conditions $n(j) = n(j')$, $p(j) \neq p(j')$, $i(j)$ or $i(j') \in \boldsymbol{TSC}$, and $j < j'$.

In F1, each slot has exactly one real cargo. In contrast, F2 and F3 use more slots than the number of real cargos. Therefore, as we saw earlier, some slots may go empty in F2 and F3. To eliminate redundant permutations involving empty slots, we push such slots to the beginning of the schedule rather than leave them dispersed among the non-zero slots. Therefore, we force each slot preceding an empty slot to be empty as well by using,

$$\sum_j x_{j(k+1)} \geq \sum_j x_{jk} \qquad k < K_p, j \in \boldsymbol{J}_p \tag{3.27}$$

For the pair-wise sequencing approach, we notice that some sequences can be deduced from other sequences. For example, if $y_{12} = 1$ and $y_{23} = 1$, then we know that $y_{13} = 1$. That is, if cargo 2 transfers after cargo 1, cargo 3 transfers after cargo 2, then cargo 3 must transfer after cargo 1. To this end, for F1, we have,

$$y_{jj''} + 1 \geq y_{jj'} + y_{j'j''} \qquad p(j) = p(j') = p(j''),\ n(j) \neq n(j') \neq n(j''), j < j' < j'' \tag{3.28a}$$

$$y_{jj''} \leq y_{jj'} + y_{j'j''} \qquad p(j) = p(j') = p(j''),\ n(j) \neq n(j') \neq n(j''), j < j' < j'' \tag{3.28b}$$

When $y_{jj'} = y_{j'j''} = 1$, $y_{jj''} = 1$ by eq. 3.28a. When $y_{jj'} = y_{j'j''} = 0$, $y_{jj''} = 0$ by eq. 3.28b. When $y_{jj'}$ or $y_{j'j''} = 0$, $y_{jj''}$ is free.

Alternatively, we can use the following equality,

$$y_{jj''} + d_{jj'j''} = y_{jj'} + y_{j'j''} \qquad p(j) = p(j') = p(j''),\ n(j) \neq n(j') \neq n(j''), j < j' < j'' \tag{3.29}$$

where, $d_{jj'j''}$ is a 0-1 continuous dummy variable. For F2 and F3, the above equation remains the same, but index ranges are $n(j) = n(j') = n(j'')$, $p(j) \neq p(j') \neq p(j'')$, $j < j' < j''$.

We did a preliminary evaluation of the effectiveness of these additional constraints on model performance and tightness. All constraints seemed generally effective in reducing solution time. Clearly, explicit sequencing of empty cargos and slots does help. Moreover, fixing some pair-wise sequences also speeded up solutions. Therefore, we add all these extra constraints (eqs. 3.26, 3.27, and 3.29) to all our formulations.

Having developed nine alternate MILP formulations addressing general chemical transshipment problems, the question now is which formulation is the best. To answer this, we study their performances using several examples in the following chapter.

CHAPTER 4

GENERAL CHEMICAL TRANSSHIPMENT OPERATIONS – MODEL EVALUATION

Number of binary variables is a good indicator of relative model performance. To this end, we first identify the factors affecting the binary variables, and develop formulations for calculating binary variables for different models. Then, we use three examples to study the application and relative performance of different models. In Example 1, we focus on the effect of travel time. Example 2 aims to study the effect of number of two-sided cargos on binary variables. Example 3 explores the effect of numbers of small or big ships on binary variables. Several useful trends are observed. In addition, an overall model ranking scheme is used to study the relative model performance. Based on the evaluations, we develop some guidelines for selecting a suitable model to use when facing different types of transshipment problems. Lastly, a heuristic using cargo aggregation method is developed to improve the model performance over large size problems.

4.1 Estimation of Binary Variables

Table 4.1 summarizes the constraints for each of the nine models for an easy reference. Clearly, it is not easy to pick one as the best model without doing a detailed numerical evaluation. However, we could get some idea about the relative performances of these models by merely comparing their numbers of binary variables. No indicator is the best or fool-proof, but fewer binary variables often lead to better performance. Of course, exceptions do exist and are often significant. Therefore, let us now compute the numbers of binary variables in F1, F2, and F3.

Table 4.1: Common, specific, and additional constraints for each model

Constraints	F11	F12	F13	F21(F31)	F22(F32)	F23(F33)
Common	3.1-3.7a, 3.12, 3.13			3.2, 3.7, 3.14, 3.15a, 3.16-3.17, 3.22-3.24		
Specific	3.7b, 3.9a,b	3.9a, 3.10	3.11a-c	3.7b, 3.15b, 3.19a,b	3.7b, 3.15b, 3.19a, 3.20	3.21a, 3.11b-c
Additional	3.26, 3.29	3.26, 3.29	3.26, 3.29	3.26, 3.27, 3.29	3.26, 3.27, 3.29	3.26, 3.27, 3.29

F1-F3 use two types of binary variables. One type (x-variables) is for the slot-based sequencing, while the other type (y-variables) is for the pair-wise sequencing. The former for models F1, F2, and F3 are:

$$\text{F1: } \sum_n |\boldsymbol{IS}_n||\boldsymbol{JS}_n| \tag{4.1a}$$

$$\text{F2: } \sum_p |\boldsymbol{J}_p|^2 \tag{4.1b}$$

$$\text{F3: } \sum_m K_m |\boldsymbol{JB}_m| \tag{4.1c}$$

While, the latter are:

$$\text{F1: } \frac{1}{2}\sum_p \left[|\boldsymbol{J}_p|(|\boldsymbol{J}_p|-1) - \sum_n |\boldsymbol{J}_p \cap \boldsymbol{JS}_n|(|\boldsymbol{J}_p \cap \boldsymbol{JS}_n|-1) \right] \tag{4.2a}$$

$$\text{F2 \& F3: } \frac{1}{2}\sum_n \left[|\boldsymbol{JS}_n|(|\boldsymbol{JS}_n|-1) - \sum_p |\boldsymbol{J}_p \cap \boldsymbol{JS}_n|(|\boldsymbol{J}_p \cap \boldsymbol{JS}_n|-1) \right] \tag{4.2b}$$

Because the number of binary variables should be a good indicator of the relative performances of F2 and F3, let us compare them first. We see from eqs. 4.1b,c and 4.2b that F2 and F3 differ only in the numbers of x-variables. Now, consider a big ship m with a j-cargos on one side and b j-cargos on the other. With no loss of generality, let $a \geq b$. Then, the number of x-variables for m in F2 is $a^2 + b^2$ and that in F3 is $a(a+b)$. Thus, the number of extra binary variables in F3 is $b(a-b) \geq 0$ and F3 always uses more binary variables than F2. Therefore, we expect F3 to perform worse than F2.

Although the number of binary variables is a good indicator of model performance in many cases, many other factors such as the types of binary variables, the number of big-M constraints, etc. also affect computation time. Therefore, in the next section, we perform a detailed numerical comparison of various models using several examples.

4.2 Examples

For illustrating the application of our nine models and comparing their performances, we use three examples of varying sizes. Example 1 aims to illustrate the effect of (small) ship travel times. Examples 2 and 3 explore the factors that affect the numbers of binary variables. Specifically, Example 2 focuses on the effect of the number of two-sided cargos, while Example 3 investigates the effect of numbers of big/small ships on the model performance.

We solved all three examples using CPLEX 9.0 with GAMS version 21.7 on a Pentium IV computer with 1.0 GB of RAM and Windows XP (SP2). All solutions have 0.0% gap, i.e. they are optimal. Note that the big-M constraints are present in all nine models. The value of big-M (H) can affect the model performance greatly and erratically. Thus, consistency in selecting the big-M values is a critical issue. We vary the big-M value from example to example, but we keep it identical for all models on the same problem to ensure a fair and consistent comparison. In Examples 1a-b, we use $H = 120$ and $H = 80$ respectively. In Examples 2a-c and 3a-c, we use $H = 100$. We selected these values as roughly twice the departure time of the last vessel.

For all examples, we use Gantt charts to present transshipment schedules. We use three shaded rectangles to represent slots with different activities of ships. For example, a white rectangle represents idling, grey represents service (transfer), and black represents travel from position to position. We tag each grey rectangle with an index for the cargo that is transferred in that slot. The numbers underneath rectangles denote the start/end times of activities. A carrier departs at the end of its last slot. We use one row of rectangles for each small ship and each position of each big ship. Although the two positions have common arrival/departure times, their activities are different and independent.

4.2.1 Example 1

We first consider a simple transshipment problem involving only one big ship ($m1$; $M = 1$) with thirteen transshipment cargos ($i1$ to $i13$; $I = 13$), one ($i3$) of which is a two-sided cargo. Four small ships ($n1$ to $n4$; $N = 4$) are to receive these cargos. Tables 4.2-4.3 list all the data for this example including expected arrival times, time-charter costs, cargo service times, etc. We assume that the travel time from any position to any other is the same (3 h) for all small ships. The arrival time of $m1$ is 1.2 h, while those of $n1$ to $n4$ are 1.2 h, 1.2 h, 0.0 h, and 6.0 h respectively.

All nine models give the same optimal transshipment schedule (Figure 4.1) with a total time-charter cost of $17823.80. Figure 4.1 shows that $n1$ and $n3$ start transshipments first. $n2$ waits until $n1$ finishes service in position $p2$. This is because $n2$ has the lowest time-charter cost among the three ($n1$, $n2$, and $n3$) ships that arrive first and compete for service.

Table 4.2: Expected arrival times and time-charter costs of carriers for the examples

Carrier	ETA (h)	Time-charter cost ($/h)
Examples 1a-b		
*m*1	1.2	160
*n*1	1.2	85
*n*2	1.2	70
*n*3	0.0	105
*n*4	6.0	101
Examples 2a-c		
*m*1	0.0	150
*m*2	0.0	180
*m*3	0.0	250
*n*1	0.0	90
*n*2	0.0	75
*n*3	0.0	105
*n*4	0.0	80
Examples 3a-c		
*m*1	2.0	150
*m*2	2.0	210
*m*3	1.5	160
*n*1	0.0	90
*n*2	0.0	75
*n*3	2.5	105
*n*4	2.0	65
*n*5	4.0	80

Table 4.3: Deep-sea carriers, short-sea carriers, unloading positions, and service times of cargos for Examples 1a-b and 2a-c

Cargo (i)	Deep-sea carrier (m)	Short-sea carrier (n)	Unloading position (p)	Service time (h)
Examples 1a-b				
1	1	1	1	9.0
2	1	2	2	4.5
3	1	2	1, 2	6.5
4	1	3	1	5.0
5	1	3	1	3.5
6	1	3	2	11.9
7	1	1	2	6.2
8	1	2	1	6.6
9	1	3	2	8.5
10	1	1	1	6.2
11	1	2	1	4.6
12	1	4	1	5.6
13	1	4	2	3.5
Examples 2a-c				
1	1	1	1	6.0
2	1	1	2	4.2
3	1	3	2	6.7
4	1	2	1	4.7
5	1	2	1	3.7
6	1	1	2	11.7
7	1	2	2	6.2
8	1	3	1	6.5
9	2	4	3	8.0
10	2	4	4	6.0
11	2	3	4	3.9
12	2	2	3	8.7
13	3	3	5	3.9
14	3	1	5	3.7
15	3	4	6	3.5

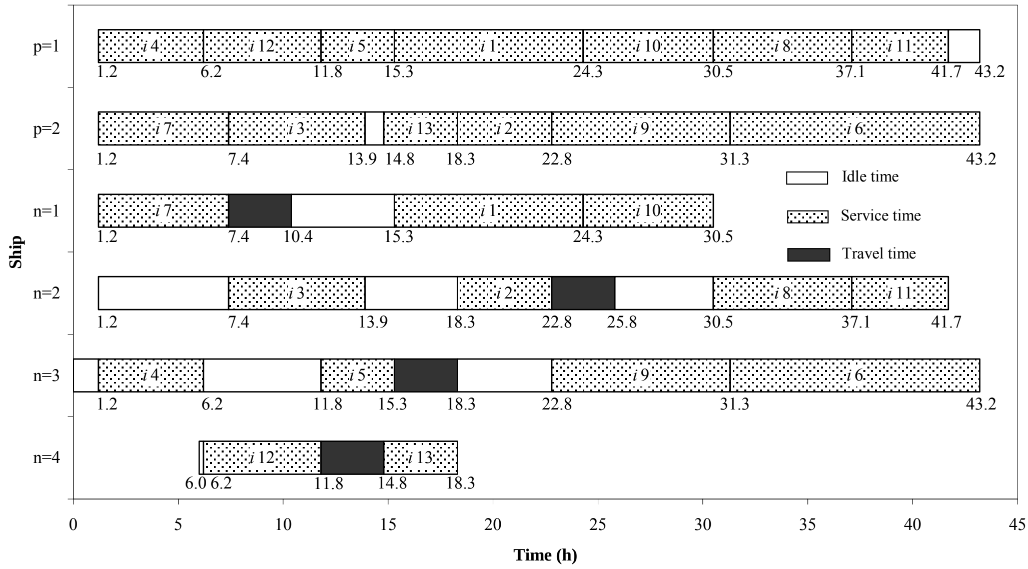

Figure 4.1: Gantt chart for the optimal transshipment schedule of Example 1a

In practice, it takes some time for a small ship to move from one location to another. Though it may be only a few minutes or hours, this travel time affects the optimal schedule. Neglecting travel times results in sub-optimal schedules. To illustrate this point, we solved this example assuming zero travel times. We call this as Example 1b. Although the new transshipment schedule is obviously different, we use its service sequences to compute the total time-charter cost, while accounting for the actual travel times. The new total time-charter cost is \$18877.90, which is 5.91% higher than \$17823.80 for Example 1a. This clearly illustrates that zero travel times is not a good assumption in transshipment problems.

Table 4.4 lists the results of and model statistics for Examples 1a-b. F1 models use 85 binary variables, F2 models use 113, and F3 models use 125. F11-F12 perform the best in Example 1a, which seems to correlate well with their fewer binary variables. However, surprisingly, F21, F22, and F31 (F23, F33 are worse than all F1 models; F32 is worse than F11 and F12) outperform F1 in Example 1b even with many more binary variables. F13, F23, and F33 always give the highest rMIP objectives, which is consistent with our expectation of their relative tightness.

However, as we expected before, in spite of being tighter, they are the worst performers. F3 requires more binary variables (125 vs. 113), continuous variables, constraints, and nonzeros than F2. Therefore, as expected, it is slower than F2 and has the same rMIP value as F2. Furthermore, it shows no advantage over any other models, so we do not include F3 in our subsequent evaluation.

Now, we address the factors that may affect the binary variables and hence model performance. From section 4.1, we know that the number of slots, the number of cargos in each carrier (deep-sea/short-sea), the numbers of carriers, the number of two-sided cargos, the assignment of cargos to deep-sea/short-sea carriers, etc. directly affect the number of binary variables. Therefore, we will now vary these factors in the following examples.

Table 4.4: Model statistics and computational results for Examples 1a-b

Example 1a ($M = 1, N = 4, I = 13, \|\boldsymbol{TSC}\| = 1$)						
	F11	F12	F13	F21	F22	F23
(x+y)-variables	49+36	49+36	49+36	100+13	100+13	100+13
Continuous variables	128	128	177	52	52	152
Constraints	362	303	326	413	315	313
Non zeros	1338	1204	1253	1670	1512	1470
Nodes	644039	1251732	1203964	699285	1269667	1089744
rMIP objective ($)	12076.00	12076.60	12076.60	9696.10	11827.62	11859.63
MIP objective ($)	17823.80	17823.80	17823.80	17823.80	17823.80	17823.80
Integrality gap (%)	32.2	32.2	32.2	45.6	33.6	33.5
CPU time (s)	458	653	991	731	1292	3226
	F31	F32	F33			
(x+y)-variables	112+13	112+13	112+13			
Continuous variables	54	54	166			
Constraints	455	345	343			
Non zeros	1852	1672	1628			
Nodes	798526	2284452	1248252			
rMIP objective ($)	9696.10	11827.62	11859.63			
MIP objective ($)	17823.80	17823.80	17823.80			
Integrality gap (%)	45.6	33.6	33.5			
CPU time (s)	854	2269	3970			
Example 1b ($M = 1, N = 4, I = 13, \|\boldsymbol{TSC}\| = 1, \theta_{npp'} = 0$)						
	F11	F12	F13	F21	F22	F23
(x+y)-variables	49+36	49+36	49+36	100+13	100+13	100+13
Continuous variables	128	128	177	52	52	152
Constraints	362	303	326	413	315	313
Non zeros	1320	1186	1235	1670	1512	1470
Nodes	2241186	3986000	4358482	331160	1410561	856204
rMIP objective ($)	12076.60	12076.60	12076.60	9696.10	12126.93	12170.39
MIP objective ($)	17021.40	17021.40	17021.40	17021.40	17021.40	17021.40
Integrality gap (%)	29.1	29.1	29.1	43.0	28.8	28.5
CPU time (s)	1768	1760	2848	352	1407	3059
	F31	F32	F33			
(x+y)-variables	112+13	112+13	112+13			
Continuous variables	54	54	166			
Constraints	455	345	343			
Non zeros	1852	1672	1628			
Nodes	609880	1530816	1285249			
rMIP objective ($)	9696.10	12126.93	12170.39			
MIP objective ($)	17021.40	17021.40	17021.40			
Integrality gap (%)	43.0	28.8	28.5			
CPU time (s)	751	1816	4607			

4.2.2 Example 2

In this example, we select a more complex situation involving three deep-sea carriers ($m1$ to $m3$; $M = 3$), four short-sea carriers ($n1$ to $n4$; $N = 4$), and fifteen transshipment cargos ($i1$ to $i15$, $I = 15$). Tables 4.2-4.3 give the required data for Examples 2a-c. The travel time between the two sides of any deep-sea carrier is 2 h, between $m1$ and $m2$ is 3 h, between $m2$ and $m3$ is 3 h, and between $m1$ and $m3$ is 4 h. Example 2a has only one-sided cargos. Then, we gradually increase the number of two-sided cargos by converting some one-sided cargos. The two-sided cargos are $i1$, $i8$, and $i13$ in Example 2b, and $i1$, $i2$, $i6$, $i7$, $i8$, $i10$, and $i13$ in Example 2c. These examples also simulate congestion problems that arise because of all carriers arriving at the transfer port at the same time. In such a situation, it is hard even for an experienced scheduler to decide which cargo should start transshipment first.

Figures 4.2 and 4.3 show the optimal transshipment schedules for examples 2a and 2c respectively. Example 2a with no two-sided cargos has a total time-charter cost of \$22099.00. When there are seven two-sided cargos as in Example 2c, the total time-charter cost reduces to \$21143.00. This is lower, because two-sided cargos give more flexibility in operations. Comparing the schedules of Examples 2a and 2c, we see that Example 2c has a more compact schedule with less waiting time for big ships than example 2a. In addition, the travel times of ships $n2$ and $n3$ are shorter. In Example 2a, $n2$ travels from position $p2$ to $p1$ of $m1$ to receive cargos $i4$, $i5$, and $i7$. However, when $i7$ becomes a two-sided cargo, $n2$ receives $i4$, $i5$, and $i7$ from $p1$ and saves the travel time from $p2$ to $p1$. This results in a better schedule.

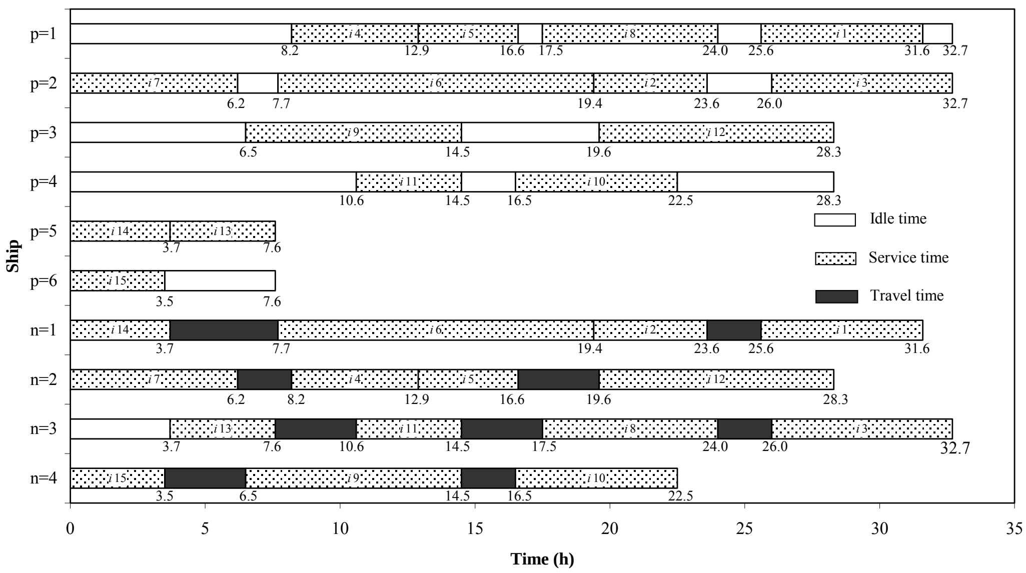

Figure 4.2: Gantt chart for the optimal transshipment schedule of Example 2a

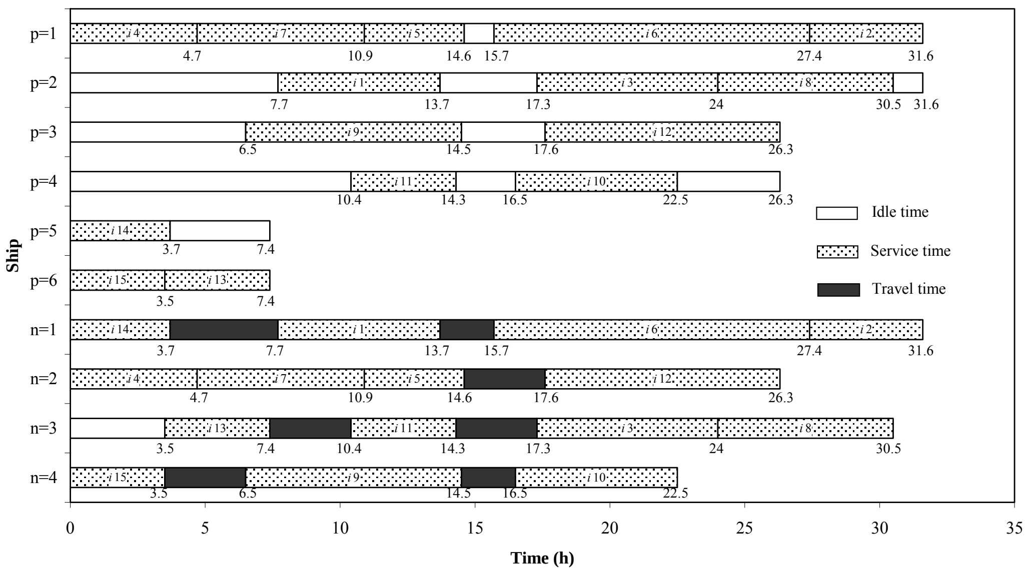

Figure 4.3: Gantt chart for the optimal transshipment schedule of Example 2c

Table 4.5 shows the results of and model statistics for Examples 2a-c. As the number of two-sided cargo increases, so do $|\boldsymbol{JS}_n|$ and $|\boldsymbol{J}_p|$, but not $K_n = |\boldsymbol{IS}_n|$. Thus, the number of slots for each small ship *n* remains the same, but the numbers of cargo pairs for each ship *n* and position increase, and those with the same origin and destination ships remain the same or increase. Thus, the *x*-variables in both models increase (eqs. 4.1a-b); while the *y*-variables either remain constant or increase with more two-sided cargos (eqs. 4.2a-b). However, because F2 has more slots, the *x*-variables increase much more than in F1. Therefore, an increase in two-sided cargos has a greater impact on F2 than F1. The model statistics for Examples 2a-c confirm these conclusions. The binary variables increase by 45, when we go from 0 to 7 two-sided cargos in F1, which increases the solution time almost a hundred fold. For F2, the impact is more severe. The binary variables increase by 83 and F2 fails to solve Example 2c even after 5000 s.

Table 4.5: Model statistics and computational results for Examples 2a-c

	F11	F12	F13	F21	F22	F23
Example 2a (M = 3, N = 4, I = 15, $\lvert \boldsymbol{TSC} \rvert$ = 0)						
(x+y)-variables	57+13	57+13	57+13	45+19	45+19	45+19
Continuous variables	179	176	233	68	68	110
Constraints	369	298	327	333	294	288
Non zeros	1332	1155	1215	1030	925	877
Nodes	3763	5572	1988	59107	48864	58848
rMIP objective ($)	11952.50	14060.50	14083.71	12275.00	13515.15	13544.36
MIP objective ($)	22099.00	22099.00	22099.00	22099.00	22099.00	22099.00
Integrality gap (%)	45.9	36.4	36.3	44.5	38.8	38.7
CPU time (s)	3.0	3.3	2.4	31.4	24.2	52.0
Example 2b (M = 3, N = 4, I = 15, $\lvert \boldsymbol{TSC} \rvert$ = 3)						
(x+y)-variables	69+20	69+20	69+20	68+29	68+29	68+29
Continuous variables	223	223	292	102	102	170
Constraints	442	359	388	469	407	401
Non zeros	1699	1489	1558	1590	1422	1368
Nodes	33604	84155	23800	626513	809543	570895
rMIP objective ($)	11902.50	13478.50	13480.80	11762.50	12567.41	12583.19
MIP objective ($)	21998.00	21998.00	21998.00	21998.00	21998.00	21998.00
Integrality gap (%)	45.9	38.7	38.7	46.5	42.9	42.8
CPU time (s)	29	65	36	457	711	907
Example 2c (M = 3, N = 4, I = 15, $\lvert \boldsymbol{TSC} \rvert$ = 7)						
(x+y)-variables	84+31	84+31	84+31	106+41	106+41	106+41
Continuous variables	255	255	339	141	141	247
Constraints	542	440	473	656	556	550
Non zeros	2103	1844	1928	2376	2102	2036
Nodes	298616	1177240	269554	2829184	2433350	1183737
rMIP objective ($)	11152.50	12126.53	12134.31	10643.50	11062.86	11085.36
MIP objective ($)	21143.00	21143.00	21143.00	21143.00	21143.00	21143.00
Integrality gap (%)	47.3	42.6	42.6	49.7	47.7	47.6
CPU time (s)	383	1067	441	5000	5000	5000
Best possible				17819.98	17627.35	17661.00
Relative gap (%)				15.72	16.63	16.47

4.2.3 Example 3

In this example, we study the effect of varying the numbers of carriers. For Example 3a, we use two deep-sea carriers (M = 2), four short-sea carriers (N = 4), and sixteen one-sided transshipment cargos (I = J = 16). Then, we change the number of short-sea carriers in Example 3b, and the number of deep-sea carriers in Example 3c, while keeping the other parameters unchanged.

Tables 4.2 and 4.6 present the data for these examples. We assume the travel time between the two positions of each deep-sea carrier as 2 h, between *m*1 and *m*2 as 2.5 h, between *m*2 and *m*3 as 2.5 h, and between *m*1 and *m*3 as 3 h. Table 4.7 gives the results and model statistics, while Figures 3.4, 4.4, and 4.5 display the optimal schedules. In Example 3b, we include a new small ship (*n*5) with a higher time-charter cost than *n*4 (80 vs. 65). In Example 3c, we add a new big ship (*m*3) with a higher cost than *m*1 (160 vs. 150). It is not surprising to observe that the optimizer gives higher priorities to these more expensive ships and ends their operations first in the optimal schedules. For instance in Example 3b, *p*4 ($ETAB_{m2}$ = 2.0 h) waits for *n*5 ($ETAS_{n5}$ = 4.0) to finish operations, even though both *p*4 and *n*4 arrive earlier ($ETAS_{n4}$ = 2.0). Again, this agrees with the observation from Example 1a.

Table 4.6: Deep-sea carriers, short-sea carriers, unloading positions, and service times of cargos for Examples 3a-c

Cargo (*i*)	Deep-sea carrier (*m*)		Short-sea carrier (*n*)		Unloading position (*p*)		Service time (h)
	(a & b)	(c)	(a & c)	(b)	(a & b)	(c)	
1	1	1	1	1	1	1	9.0
2	1	1	1	1	2	2	7.3
3	1	1	3	3	2	2	9.8
4	1	1	2	2	1	1	7.8
5	1	1	2	2	1	1	6.8
6	1	1	1	1	2	2	14.8
7	1	1	2	2	2	2	9.3
8	1	1	3	3	1	1	9.6
9	2	2	4	4	3	3	11.0
10	2	2	4	4	4	4	9.2
11	2	2	3	3	4	4	7.0
12	2	2	2	2	3	3	11.7
13	2	2	3	3	3	3	7.0
14	2	3	1	1	3	5	6.7
15	2	2	4	5	4	4	6.5
16	1	3	2	2	2	6	7.8

Table 4.7: Model statistics and computational results for Examples 3a-c

	F11	F12	F13	F21	F22	F23
Example 3a (M = 2, N = 4, I = 16, |***TSC***| = 0)						
(x+y)-variables	66+21	66+21	66+21	66+21	66+21	66+21
Continuous variables	180	180	246	65	65	131
Constraints	399	321	349	389	327	323
Non zeros	1565	1383	1449	1345	1261	1213
Nodes	44047	209221	80248	751885	857825	997473
rMIP objective ($)	20070.01	21558.38	21566.80	19277.50	22188.95	22249.37
MIP objective ($)	29829.00	29829.00	29829.00	29829.00	29829.00	29829.00
Integrality gap (%)	32.7	27.7	27.7	35.4	25.6	25.4
CPU time (s)	35	145	118	556	594	1979
Example 3b (M = 2, N = 5, I = 16, |***TSC***| = 0)						
(x+y)-variables	62+22	62+22	62+22	66+20	66+20	66+20
Continuous variables	177	177	239	66	66	132
Constraints	389	316	343	403	341	337
Non zeros	1515	1345	1407	1373	1289	1241
Nodes	48508	336007	71274	317265	628011	537370
rMIP objective ($)	18802.51	20534.63	20545.88	19797.50	23344.40	23399.16
MIP objective ($)	30423.00	30423.00	30423.00	30423.00	30423.00	30423.00
Integrality gap (%)	38.2	32.5	32.5	34.9	23.3	23.1
CPU time (s)	38	230	126	221	491	1081
Example 3c (M = 3, N = 4, I = 16, |***TSC***| = 0)						
(x+y)-variables	66+15	66+15	66+15	52+22	52+22	52+22
Continuous variables	204	204	270	69	69	121
Constraints	406	328	356	364	318	312
Non zeros	1610	1428	1494	1189	1133	1085
Nodes	3674	7131	8809	116092	126992	33352
rMIP objective ($)	21285.75	23683.50	23687.71	17948.50	20740.00	20778.03
MIP objective ($)	29972.50	29972.50	29972.50	29972.50	29972.50	29972.50
Integrality gap (%)	29.0	21.0	21.0	40.1	30.8	30.7
CPU time (s)	3.3	5.8	13.0	78.6	80.9	60.3

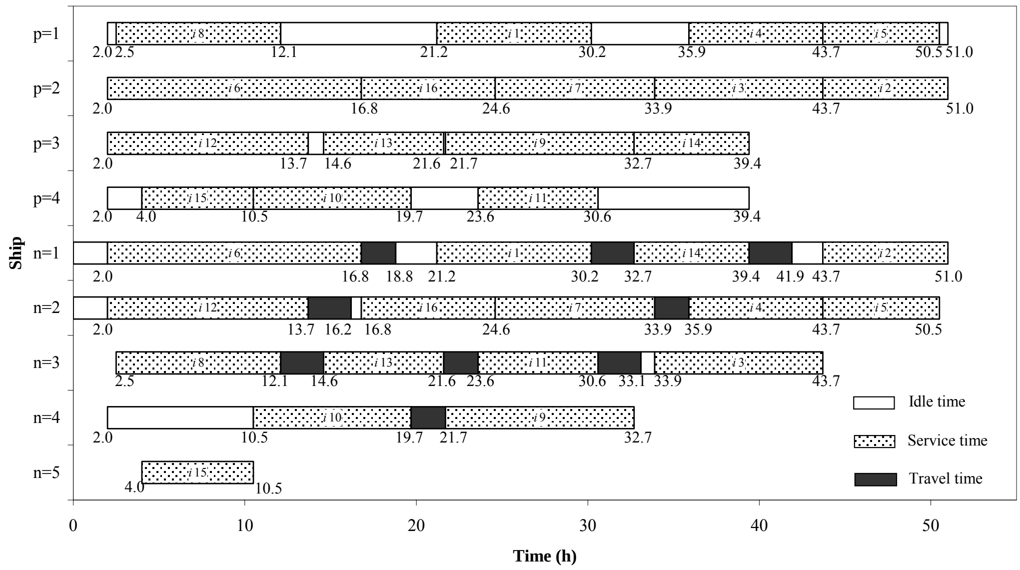

Figure 4.4: Gantt chart for the optimal transshipment schedule of Example 3b

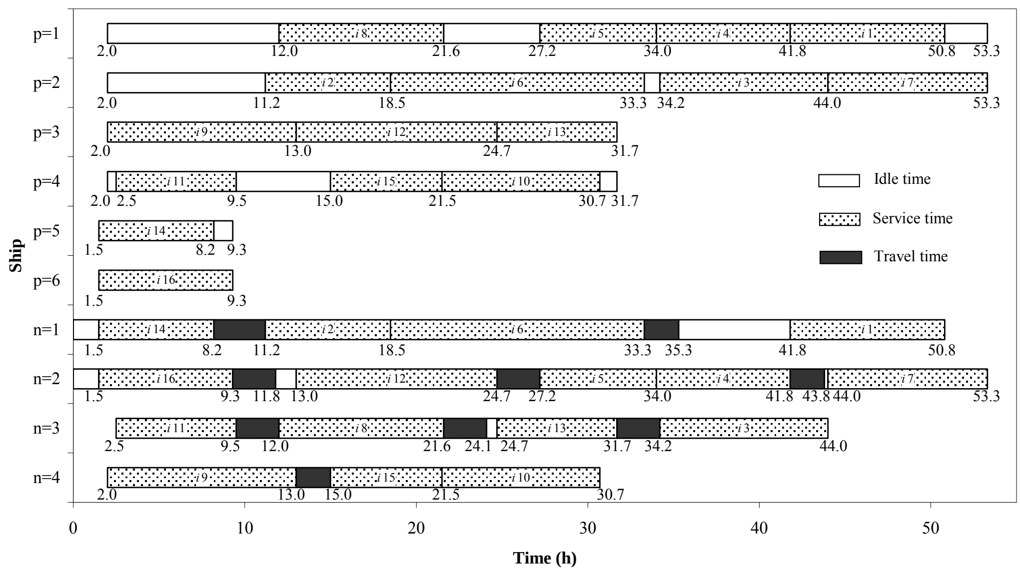

Figure 4.5: Gantt chart for the optimal transshipment schedule of Example 3c

The addition of *n*5 in Example 3b reduces three binary variables in F1 and one in F2. In Example 3c, the addition of *m*3 reduces six binary variables in F1 and thirteen F2. The above reductions occur, because the number of cargos is the same for Examples 3a-c. When N increases, $K_n = |\boldsymbol{IS}_n|$ and $|\boldsymbol{JS}_n|$ reduce in F1, and hence the *x*-variables reduce. Although $|\boldsymbol{J}_p|$ remains unchanged, the number of cargos with the same origin and destination ships may decrease in F1, hence the *y*-variables may increase. Thus, the *x*-variables and *y*-variables change in opposite directions, but the change in the former outweighs that in the latter. As a result, the binary variables reduce in F1. The reverse argument applies to F2, where the *x*-variables remain constant, but the *y*-variables may decrease. F3 has the same trend. In summary, the net effect for all models is a reduction in binary variables. Although the number of binary variables is not a fool-proof indicator of performance, these changes allow us to get some insight into the effects of different binary variables (*x* vs. *y*) on model performance, as we discuss next.

4.2.4 Observations

Based on the eight examples (1a-b, 2a-c, and 3a-c), we extract three useful observations.

The first concerns the effect of the two types of binary variables, *x*-variables for the slot-based modeling approach and *y*-variables for the sequence-based modeling approach. From the model statistics and computational results of Examples 1-3, we find that fewer binary variables generally mean faster solution. However, as highlighted earlier, exceptions do exist. For instance, in Examples 1a and 2b, F1 models have roughly the same numbers (85 vs. 89) of binary variables. Yet, F1 needs over 400 s for Example 1a and less than 100 s for Example 2b. Similarly, F2 solves Example 1b much faster than F1, though F2 has many more binary variables (113 vs. 85). Similar exceptions occur in Examples 2a and 3c. We believe that this may be because the two different types of binary variables affect model performance differently. Pitty (2004) showed that the sequence-based modeling approach is inferior to the slot-based approach in most sequencing problems, and this is largely due to the presence of big-M constraints. This observation certainly seems to hold in this problem, as we can see in the following instances. F1 models use more *y*-

variables (36 vs. 20) in Example 1a than Example 2b. The solution time is more than 400 s in F1, while less than 100 s in F2. In Example 1b, F1 models have 36 y-variables and all take more than 1700 s, while F2 models have only 13 and F21 takes less than 400 s, F22 takes 1407 s, and F23 takes 3059 s. In Example 2a, F1 models have 13 y-variables and take less than 4 s, while F2 models have 19 and take more than 20 s. In Example 3c, F1 models have 15 y-variables, while F2 models have 22. F1 models take less than 13 s, while F2 models take more than 60 s. In all of these examples, it seems that the ill-effect of the sequence-based modeling overpowers the advantage of having fewer binary variables.

Our second observation relates to the effect of formulation tightness. Of the three alternative formulations for each model, the third alternative (Fx3 models) always has the largest rMIP objective value, and hence is the tightest formulation. In contrast, the first alternative has the smallest rMIP objective value. We had expected this and the reason for this is obvious. The first alternative has two big-M constraints, while the third has none. However, as we expected earlier, the tighter formulations, namely the third alternatives, are slower than the first in all examples. The benefit of fewer big-M constraints is indeed offset by the increase in continuous variables. Furthermore, it is interesting that even the equality constraint that we used to replace one big-M constraint does not improve performance, even though it does not increase the number of continuous variables. We are not sure why this is so, although it makes sense intuitively. Furthermore, it is also surprising that the addition of the equality constraint to Fx1 models also does not improve performance, although it should make them tighter without increasing problem size.

Our third observation relates to the use of the additional constraints. Earlier, we stated that the inclusion of the additional constraints helps reduce the solution time.

We base our observation on the following results on F11 and F21 for Examples 1a and 2b. Table 4.8 compares the model statistics for the two examples with and without the additional constraints. The additional constraints reduce solution times in most cases, especially for the larger problems. In Example 1a, this reduction is from 924 s to 458 s for F11, and from 1095 s to 731 s for F21, which are significant improvements. Thus, we conclude that our proposed additional constraints in section 5 to sequence empty slots, cargos, and cargo-cargo pairs are quite effective.

Table 4.8: Model statistics and computational results for selected examples with and without additional constraints

Example 1a				
	F11*	F11	F21*	F21
(x+y)-variables	49+36	49+36	100+13	100+13
Continuous variables	79	128	48	52
Constraints	313	362	397	413
Non zeros	1142	1338	1482	1670
Nodes	1517954	644039	1276616	699285
rMIP objective ($)	12076.60	12076.60	9696.10	9696.10
MIP objective ($)	17823.80	17823.80	17823.80	17823.80
Integrality gap (%)	32.2	32.2	45.6	45.6
CPU time (s)	924	458	1095	731
Example 2b				
	F11*	F11	F21*	F21
(x+y)-variables	69+20	69+20	68+29	68+29
Continuous variables	202	223	62	102
Constraints	421	442	417	469
Non zeros	1615	1699	1330	1590
Nodes	27918	33604	876518	626513
rMIP objective ($)	11902.50	11902.50	11762.50	11762.50
MIP objective ($)	21998.00	21998.00	21998.00	21998.00
Integrality gap (%)	45.9	45.9	46.5	46.5
CPU time (s)	26	29	562	457

* Base formulation: common + specific constraints

4.2.5 Overall Model Ranks

It is clear that none of the six models has performed the best for all examples. To get an overall ranking of the models based on our limited numerical evaluation, we proceed as follows.

One common way to compare models (m) is to compute their mean relative CPU times on several problems (p). We do this by computing a relative CPU time ($RCPU_{mp}$) for each model m on each problem p as follows:

$$RCPU_{mp} = \frac{CPU_{mp}}{\min_{p}\left[CPU_{mp}\right]} \tag{4.3}$$

Gupta and Karimi (2003) and Karimi et al. (2004) used these relative CPU times to compute an arithmetic mean over all problems for each model. However, it seems that geometric mean is a superior measure than arithmetic mean, as it gives more reliable ranking. Therefore, we define geometric mean rank as,

$$GMR_m = \sqrt[p]{\prod_{p} RCPU_{pm}} \tag{4.4}$$

The only problem with the above measure is that we still do not have a way to take into account the effect of different problem sizes. Generally, we would rank a model higher, if it performs better on larger problems. For this reason, we compute a weighted arithmetic mean of relative CPU times ($WRCPU_m$) as well.

$$WRCPU_m = \frac{\sum_{p} BV_{pm} RCPU_{pm}}{\sum_{p} BV_{pm}} \tag{4.5}$$

However, since we have both x-variables and y-variables and they affect the performance differently, we compute two $WRCPU_m$s, one with x-variables as weights and the other with y-variables. Then take the geometric mean of these two means get an overall weighted mean rank WMR_m.

Table 4.9 lists the two mean ranks (GMR_m and WMR_m) for the six models. F11, F12, and F13 rank higher than F21, F22, and F23. Therefore, a slot-based approach for the small ships and a sequence-based approach for the big ships seems a better overall approach. Among the three F1 and F2 models, F11 and F21 rank first respectively. Although F2 models perform poorer than F1 models in general, they seem to do better for fewer deep-sea carriers. As discussed earlier, this is probably due to the effect of y-variables.

Table 4.9: Mean ranks for models

	GMR_m*	WMR_m
F11	1.12	1.58
F12	1.64	3.16
F13	1.55	3.08
F21	2.62	10.75
F22	3.25	13.81
F23	4.46	24.65

*Note: Runs that did not yield optimal solutions within 5000 s were not included in computing the mean CPU times.

4.3 Cargo Aggregation Heuristic

A multi-compartment carrier may have up to 15 compartments. However, several cargos may come from one compartment or one ship. Therefore, a real transshipment operation may easily involve tens of cargos. We used up to sixteen cargos in our illustrative examples and the computational performance was quite good. Our proposed models solved even examples with hundreds of binary variables within half an hour. However, it is practically useful to have a model that can solve much larger problems efficiently, even if it may not guarantee optimality.

We achieve this by exploiting the fact that several cargos may involve the same ships (small and big) in practice. In fact, most cargos involving the same ships transship consecutively in the optimal schedules. For instance, cargos $i9$ and $i6$ are consecutive in Example 1a, *i*4 and *i*5 in Example 2a, and so on. This makes sense because a small ship saves travel (and other) time by serving the cargos from a ship in one shot. In fact, one may very well prefer this in practice, because a small ship may require some other resources or activities to travel from one position to another, which our model does not consider. Therefore, it seems reasonable to aggregate cargos with the same unloading position and designated short-sea carrier into one single one-sided cargo with a service time equal to the sum of the service times of individual cargos. In fact, doing this may even result in a service time that is less than the sum, because a small ship may not need to repeat some activities associated with the transshipment of each cargo. This aggregation of cargos would reduce the effective number of cargos and result in smaller problems.

To test the effectiveness of this cargo aggregation heuristic, we redefine the cargos for all examples using the above assumption and resolve them. Note that we cannot aggregate two-sided cargos, since we cannot fix their unloading positions a priori. Tables 4.10-4.12 show the results and model statistics for the revised examples using the heuristic. The binary variables reduce significantly in all examples. For instance, they reduce from 85 to 42 in F1 and 113 to 57 in F2 for Example 1a. As expected, the model performance improves greatly. The aggregation heuristic allows us to solve most examples within seconds. Even those (Examples 1a-b and 3a-b) that took thousands of seconds previously gave optimal solutions within minutes. For instance, F11 in Example 1a took only 0.64 s compared to 458 s for the optimal solution. Of course, we do get suboptimal solutions for some examples, but the deviations from the optima are quite small. The greatest deviation of 2.23% from the optimal occurs in Example 3a, while we get optimal solutions for Examples 2a-c. Hence, we conclude that the cargo aggregation heuristic is extremely effective in solving larger problems of practical interest without significantly sacrificing the quality of solution. However, note that this heuristic will not be very effective, when much cargo aggregation may not be possible due to the presence of many two-sided cargos.

Table 4.10: Model statistics and computational results for Examples 1a-b using the heuristic assumption

Example 1a ($M = 1$, $N = 4$, $I = 13$, $\lvert \boldsymbol{TSC} \rvert = 1$)						
	F11	F12	F13	F21	F22	F23
(x+y)-variables	24+18	24+18	24+18	50+7	50+7	50+7
Continuous variables	71	71	95	39	39	89
Constraints	199	169	184	248	200	198
Non zeros	677	614	638	900	830	800
Nodes	1478	2467	1046	1570	2062	1198
rMIP objective ($)	12076.60	12076.60	12076.60	11437.60	12515.39	12597.34
MIP objective ($)	17836.90	17836.90	17836.90	17836.90	17836.90	17836.90
Integrality gap (%)	32.3	32.3	32.3	35.9	29.8	29.4
CPU time (s)	0.64	0.97	0.59	1.02	1.28	2.03
CPU time reduction (%)*	99.86	99.85	99.94	99.86	99.90	99.94
Sub-optimality (%)**	0.07	0.07	0.07	0.07	0.07	0.07
Example 1b ($M = 1$, $N = 4$, $I = 13$, $\lvert \boldsymbol{TSC} \rvert = 1$, $\theta_{npp'} = 0$)						
(x+y)-variables	24+18	24+18	24+18	50+7	50+7	50+7
Continuous variables	71	71	95	39	39	89
Constraints	199	169	184	248	200	198
Non zeros	677	614	638	900	830	800
Nodes	1634	3073	1744	1036	2722	942
rMIP objective ($)	12076.60	12076.60	12076.60	11437.60	12515.39	12597.34
MIP objective ($)	17358.30	17358.30	17358.30	17358.30	17358.30	17358.30
Integrality gap (%)	30.4	30.4	30.4	34.1	27.9	27.4
CPU time (s)	0.55	1.00	0.80	0.73	1.55	1.67
CPU time reduction (%)	99.97	99.94	99.97	99.79	99.89	99.95
Sub-optimality (%)	1.98	1.98	1.98	1.98	1.98	1.98

* CPU time reduction (%) = (optimal CPU time – heuristic CPU time) / optimal CPU time × 100

**Sub-optimality (%) = (heuristic objective – optimal objective) / optimal objective × 100

Table 4.11: Model statistics and computational results for Examples 2a-c using the heuristic assumption

	F11	F12	F13	F21	F22	F23
Example 2a (M = 3, N = 4, I = 15, $\|\boldsymbol{TSC}\|$ = 0)						
(x+y)-variables	43+9	43+9	43+9	31+15	31+15	31+15
Continuous variables	153	150	193	60	60	88
Constraints	299	244	269	269	244	238
Non zeros	1050	915	961	776	709	667
Nodes	421	682	338	1980	2680	2983
rMIP objective ($)	12582.50	14674.14	14719.85	12653.00	13996.03	14009.65
MIP objective ($)	22099.00	22099.00	22099.00	22099.00	22099.00	22099.00
Integrality gap (%)	43.1	33.6	33.4	42.7	36.7	36.6
CPU time (s)	0.42	0.55	0.52	0.83	1.28	1.91
CPU time reduction (%)	85.82	83.60	78.72	97.36	94.70	96.34
Sub-optimality (%)	0.00	0.00	0.00	0.00	0.00	0.00
Example 2b (M = 3, N = 4, I = 15, $\|\boldsymbol{TSC}\|$ = 3)						
(x+y)-variables	54+15	54+15	54+15	50+25	50+25	50+25
Continuous variables	193	193	247	93	93	143
Constraints	364	298	323	396	352	346
Non zeros	1381	1216	1270	1280	1162	1114
Nodes	2941	3838	3199	13868	35929	36732
rMIP objective ($)	12532.50	13783.50	13783.50	12140.50	12850.00	12854.61
MIP objective ($)	21998.00	21998.00	21998.00	21998.00	21998.00	21998.00
Integrality gap (%)	43.0	37.3	37.3	44.8	41.6	41.6
CPU time (s)	2.45	2.48	4.20	8.83	24.23	35.42
CPU time reduction (%)	91.63	96.15	88.29	98.07	96.59	96.10
Sub-optimality (%)	0.00	0.00	0.00	0.00	0.00	0.00
Example 2c (M = 3, N = 4, I = 15, $\|\boldsymbol{TSC}\|$ = 7)						
(x+y)-variables	76+27	76+27	76+27	93+39	93+39	93+39
Continuous variables	236	236	312	136	136	229
Constraints	495	402	433	611	524	518
Non zeros	1910	1675	1751	2167	1930	1867
Nodes	64686	140854	59148	3468295	3248943	1408500
rMIP objective ($)	11407.50	12140.31	12155.80	10643.50	11062.43	11085.61
MIP objective ($)	21143.00	21143.00	21143.00	21143.00	21143.00	21143.00
Integrality gap (%)	46.0	42.6	42.5	49.7	47.7	47.6
CPU time (s)	66	109	102	5000	5000	5000
Best possible				19574.19	19735.63	19329.51
Relative gap (%)				7.42	6.66	8.58
CPU time reduction (%)	82.65	89.74	76.93			
Sub-optimality (%)	0.00	0.00	0.00			

Table 4.12: Model statistics and computational results for Examples 3a-c using the heuristic assumption

	Example 3a (M = 2, N = 4, I = 16, $\|\boldsymbol{TSC}\|$ = 0)					
	F11	F12	F13	F21	F22	F23
(x+y)-variables	38+13	38+13	38+13	38+13	38+13	38+13
Continuous variables	133	133	171	49	49	87
Constraints	265	219	239	261	227	223
Non zeros	995	893	931	833	793	757
Nodes	1749	3542	1222	4497	3591	2079
rMIP objective ($)	21151.50	22334.86	22334.86	20645.00	23815.53	23834.71
MIP objective ($)	30493.00	30493.00	30493.00	30493.00	30493.00	30493.00
Integrality gap (%)	30.6	26.8	26.8	32.3	21.9	21.8
CPU time (s)	1.06	1.84	1.19	2.03	1.84	2.58
CPU time reduction (%)	96.97	98.73	98.99	99.63	99.69	99.87
Sub-optimality (%)	2.23	2.23	2.23	2.23	2.23	2.23
	Example 3b (M = 2, N = 5, I = 16, $\|\boldsymbol{TSC}\|$ = 0)					
(x+y)-variables	39+15	39+15	39+15	43+13	43+13	43+13
Continuous variables	137	137	176	53	53	96
Constraints	280	233	254	297	258	254
Non zeros	1039	935	974	945	898	859
Nodes	3017	5457	1669	8393	5441	4488
rMIP objective ($)	19884.00	21553.62	21553.62	20859.50	24327.31	24402.21
MIP objective ($)	30767.00	30767.00	30767.00	30767.00	30767.00	30767.00
Integrality gap (%)	35.4	29.9	29.9	32.2	20.9	20.7
CPU time (s)	1.81	2.94	1.84	4.03	3.03	5.88
CPU time reduction (%)	95.23	98.72	98.54	98.18	99.38	99.46
Sub-optimality (%)	1.13	1.13	1.13	1.13	1.13	1.13
	Example 3c (M = 3, N = 4, I = 16, $\|\boldsymbol{TSC}\|$ = 0)					
(x+y)-variables	45+10	45+10	45+10	33+16	33+16	33+16
Continuous variables	164	164	209	56	56	89
Constraints	304	250	272	271	244	238
Non zeros	1172	1050	1095	825	798	759
Nodes	766	702	350	2049	2952	1924
rMIP objective ($)	22375.13	24341.38	24341.38	19128.50	21184.28	21249.60
MIP objective ($)	29972.50	29972.50	29972.50	29972.50	29972.50	29972.50
Integrality gap (%)	25.3	18.8	18.8	36.2	29.3	29.1
CPU time (s)	0.59	0.61	0.69	0.98	1.41	1.78
CPU time reduction (%)	82.19	89.46	94.73	98.75	98.26	97.04
Sub-optimality (%)	0.00	0.00	0.00	0.00	0.00	0.00

4.4 Conclusion

We addressed an important problem in chemical maritime transportation, where one set of multi-compartment carriers transships (ship-to-ship) liquid chemical cargos to another set of multi-compartment carriers. Using a mix of slot-based and pair-wise sequence-modeling approaches, we developed nine continuous-time MILP formulations of three basic types. A majority of the models solved moderate-size examples involving up to three donor carriers, five recipient carriers, and sixteen transshipment cargos in reasonable solution times.

Of the three types of models, F2 and F3 are quite similar and F3 with more binary variables is clearly inferior to F2. On the other hand, F1 and F2 perform differently on different types of problems. It appears that their performances improve with decreasing total and pair-wise sequencing binary variables. In general, F2 (F1) that uses the pair-wise sequencing approach for the recipient (donor) carriers seems to have fewer variables of the latter type and does better than F1 (F2), when the problem involves many recipient (donor) carriers. For the same reasons, F1 outperforms F2 in problems with many two-sided cargos.

Among the three alternate models of each type, the ones involving the big-M relaxation (convex hull relaxation) seem to be the fastest (slowest) in spite of their inferior (superior) rMIP values. Furthermore, a model ranking strategy based on geometric means of relative solution times indicates that F11 is the best model overall.

Our proposed heuristic strategy of aggregating cargos involving the same ships into single cargos reduces problem size drastically and solution times by an order of magnitude, yet gives nearly optimal (within 2.23%) solutions for the examples in this problem. This heuristic model promises to be very effective for solving large problems of practical interest. Compared to the manual procedures used in practice for such problems, our MILP models promise to reduce total operation cost by up to 6.32%.

CHAPTER 5

SCHEDULING OF TANKER LIGHTERING OPERATIONS

In this chapter, we address a special case of chemical transshipment operation – tanker lightering problem. We first describe in detail the tanker lightering problem faced by a lightering company that provides lightering services to multiple refineries within a region. Then, we present two alternate continuous-time MILP formulations for the short-term scheduling of tanker lightering operations addressing two different objectives. Our models mimic the real operations by considering many practical features. For example, we consider 2-stage lightering for very large tankers; we limit the number of simultaneous lightering to two for a tanker; we also allow the lightering vessels to pick up different crudes with a common destination refinery from different tankers in a voyage and so on.

5.1 Problem Description

A lightering company services several refineries with shallow drafts located in a region. The company operates a fleet of N non-identical, multi-compartment lightering vessels (n = 1, 2, ..., N). We simply call them vessels. M multi-compartment tankers (m = 1, 2, ..., M) with $2M$ unloading positions (p = 1, 2, ..., $2M$) are to deliver C crudes (c = 1, 2, ..., C) to the client refineries (r = 1, 2, ..., R) in the upcoming planning horizon. These crude oils vary in their origins (tankers), target destinations (refineries), densities, and heating requirements. A crude in a tanker is taken as different from the same crude in another tanker.

A tanker m may carry one or more crudes. Let $\boldsymbol{CT}_m$ denote the set of crudes that it carries, i.e. $\boldsymbol{CT}_m = \{c \mid \text{tanker } m \text{ carries crude } c\}$. Let PS_c denote the volume (m^3)

of crude c. Each tanker is destined for one or more, a priori known, refineries and faces some draft limitations. For entry into its destination refinery ports, tanker m must lighter a known weight LW_m (lightering weight in kg) of its load, which may involve one or more crude oils. Each tanker has two sides (starboard or right and larboard or left) for crude transfer, so at most two lightering vessels can offload crude from the tanker at any time; one from each side. We assume that a tanker offloads any of its crudes from either side or position.

Normally, the lightering occurs at an anchorage location. However, some large tankers require an additional lightering step in deep offshore water before they can even enter the anchorage. Therefore, we consider two lightering locations; the anchorage being closer to the refinery and the offshore being away from the refinery. If a tanker requires 2-stage lightering, then we model it by two separate tankers with different lightering locations, operations, and requirements, one lightering at the anchorage and the other at offshore. Thus, if we have two tankers, one of which requires 2-stage lightering, then we have $M = 3$ tankers with $m = m2$ and $m' = m3$ being the same physical tanker and $m2$ ($m3$) being the offshore (anchorage) tanker. Note that two indices (m and m') refer to the same physical tanker at two different locations. To facilitate this classification, we define set $\boldsymbol{TST} = \{m \mid$ the physical tanker of m is a 2-stage tanker$\}$ and set $\boldsymbol{MM} = \{(m, m') \mid$ tanker m and tanker m' respectively are the offshore and anchorage tankers of the same 2-stage physical tanker$\}$. Thus, for the previous example, both $m2 \in \boldsymbol{TST}$, $m3 \in \boldsymbol{TST}$, and $(m2, m3) \in \boldsymbol{MM}$.

The tankers arrive at their first lightering location (anchorage or offshore) at arbitrary, but known, times ($ETAB_m$) within the planning horizon. During congestion time, multiple tankers arrive in a short time interval at a transfer location and compete for lightering services. If a large tanker requires two stages of lightering, then its arrival time at the anchorage will depend on its departure time from the offshore location, as we elaborate later on. After lightering itself at the last lightering location (anchorage), each tanker travels to its destination refinery port/s to deliver the remaining crude oil.

A vessel n has N_n^U identical compartments of volume (m^3) SC_n, and it travels with velocity vf_n when loaded and velocity ve_n when empty. All lightering vessels become available at the anchorage at arbitrary, but known, times ($ETAS_n$). The operation of a vessel involves a series of voyages. Because most lightering occurs at the anchorage, we take it as the reference lightering location where a vessel starts/ends its voyages. Thus, each voyage is one full anchorage-to-anchorage round trip of a vessel, in which it visits a single refinery. A typical voyage involves the following steps. The vessel starts from the anchorage. Then, it visits one or more tankers one by one and collects one or more crudes from each tanker. These tankers could be at either anchorage or offshore, so a voyage may involve traveling between the anchorage and offshore. We assume that all crudes in a single voyage belong to a single refinery port. However, this port may vary from voyage to voyage. After collecting all its crude parcels, the vessel travels to its designated refinery port and unloads them at the refinery. Finally, it returns to the anchorage. Serving multiple tankers in one voyage is useful during periods of congestion. By doing this, a vessel saves the round-trip between the lightering location and refinery. However, it results in a longer delivery time to the refinery for the crude oil that was loaded first on the lightering vessel. Therefore, a tradeoff exists between reduced time-charter or demurrage cost for the latter tankers and longer delivery times for some crudes.

Lightering a tanker involves the following sequential tasks: mounting to the tanker or connecting to one or more compartments of the tanker one at a time via a flexible hose, pumping crude oils one at a time to different compartments of the vessel, and then dismounting from the tanker. Some crudes require heating and a vessel may not have the required heating equipment, so a vessel may not be able to carry all crudes. Therefore, we define a set $\boldsymbol{CV}_n = \{c \mid \text{vessel } n \text{ can carry crude } c\}$. Finally, we assume that a vessel does not change its transfer position, while downloading crudes during a single visit to a tanker.

The discharging operation at a refinery involves docking at the refinery port, pumping crude oils one at a time from the vessel to one or more crude tanks in the refinery, and undocking from the refinery port. Let FIN_{nc} denote the known rate at which vessel n receives crude c from a tanker, and $FOUT_n$ the rate at which it discharges crude to the refinery.

The objective of our lightering problem is to generate an optimal short-term operation schedule that minimizes the total cost for the lightering company. The schedule includes the assignment of vessels to tankers, the sequences and timings of visits to tankers, voyage details, tankers served in each voyage, the amounts and types of crude oils unloaded from each tanker, and the numbers of vessel compartments used for carrying each crude. The total cost consists of two parts: operating costs of the lightering vessels and costs related to the tankers. The former is the sum of fuel consumption and fixed operating costs. Fuel consumption is directly proportional to the total distance that a lightering vessel travels. Fixed cost is the lump-sum cost of maintenance, depreciation, and labor. We can use the cost related to the tankers as either time-charter cost or demurrage. Time charter cost is the cost paid by the client refineries to hire crude tankers on a daily basis. Within the chartered period, refineries still pay charter cost even if a tanker sits idle. Therefore, one important objective is to minimize the time tankers spend idling at some place either for lightering or for waiting. On the other hand, demurrage is the cost borne by the lightering company, when a tanker waits more than an agreed duration for lightering services. Depending on the objectives, the optimal schedules may be different. We will study the impact of different objectives on the lightering schedules later on. In addition, the arrival times of crude oils to the refineries are also measures of customer satisfaction of the refineries.

The availability of vessels often depends on the weather condition, as some vessels cannot operate during bad weather. We address this simply by removing these vessels from the list in our formulation.

Let us list all the assumptions in this problem:

(1) Effects of tides on the velocities and drafts of vessels are negligible. When a ship travels with the tide, it can move easier, so the fuel consumption reduces at the same speed. In addition, a higher tide also increases the draft of a channel/port, which may reduce the required lightering volume for a tanker.

(2) All crude parcels are compatible with each other, as far as compartment allocation is concerned. Thus, a vessel can carry a crude in any compartment without worrying about its location.

(3) The stability considerations of vessels and tankers during the lightering operations are ignored.

(4) The lightering weight LW_m of crude that a tanker m must lighter at each lightering location is fixed and known a priori.

(5) Once a tanker arrives at a lightering location (anchorage or offshore), it does not move, until it lighters the prefixed amount of crude oil for that location. In contrast, the lightering vessels may travel among different tankers.

(6) A tanker can feed at most two lightering vessels at the same time, one at each side.

(7) Any crude can be transferred from either position (side) of a tanker.

(8) A lightering vessel can receive only one crude at a time.

(9) A lightering vessel delivers crudes to only one refinery during each voyage.

(10) All compartments in each lightering vessel are identical.

(11) Anchorage is located somewhere between offshore and refineries. Therefore, the distance from offshore to a refinery is simply the sum of distances from offshore to anchorage and anchorage to the refinery.

5.2 MILP Formulation

We use two approaches for modeling the activities of lightering vessels and crude tankers over time. We use contiguous, ordered, variable-length slots for the former and use floating, unordered, variable-length task intervals (which we order using a pair-wise sequencing approach) for the latter. In the former, we assign an activity to each slot, while in the latter, each interval has a pre-assigned activity and we ensure that the activities do not clash.

The lightering of a tanker involves the transfer of varying amounts of one or more crudes by one or more lightering vessels at the same or different times. Each tanker has two sides for crude transfer. A side can serve (i.e. offload crude to) only one lightering vessel at a time. Thus, at most two vessels can lighter a tanker at any given time. Each time a vessel comes to serve a tanker, it has several choices. First, it must decide which side of the tanker it must dock. Here it has two options, larboard and starboard. Second, it must decide which of the crudes on the tanker it must transfer first. Once it has decided the crude, it must decide the amount of crude to transfer. Of these three choices, the first two are discrete in nature, while the third is continuous. To model these choices, we define several distinct batches of crudes of unknown volumes for each tanker. We call these batches as parcels. Each time a lightering vessel visits a tanker, it can select one or more of these parcels. Whenever a lightering vessel visits a tanker, it must dock on a side and then connect to a compartment of the tanker for receiving a single crude oil. We define this as the start of a new lightering parcel. This parcel ends, when the vessel terminates that particular connection. If the vessel begins another connection to receive a different crude, then another parcel begins. Thus, each parcel is a single continuous withdrawal of a single crude from one position of a tanker. It is obvious that two batches of the same crude withdrawn at different times or from different positions of the same tanker are different parcels.

In our proposed formulation, we convert each tanker into a number of distinct parcels of unknown amounts, which a vessel can select from during the entire scheduling horizon. Each parcel has three defining attributes (1) the tanker from which it is withdrawn, (2) the tanker side (larboard or starboard) from which it transfers, (3) the crude that it contains. For each tanker *m*, we postulate a maximum number of parcels that could possibly be withdrawn from that tanker by various vessels at different (but unknown) times during the scheduling horizon and we define each of these parcels fully by prefixing a crude and a transfer position for each. During a single visit, a vessel can select from at least $2|\boldsymbol{CT}_m|$ parcels. Several vessels may visit a tanker and they may visit more than once. To allow for such a possibility, we replicate these $2|\boldsymbol{CT}_m|$ parcels a desired number of times. We then assign a unique index *j* to each parcel. Let *J* be the total number of parcels (j = 1, 2, ..., J) from all tankers. Let us use an example to illustrate the generation of parcels for a tanker. In Figure 1, a tanker *m* carries two crudes. Let us replicate each parcel three times, then we get twelve possible parcels (2 crudes times 2 positions times 3 replicates) for tanker *m*. Thus, each crude can be transferred in at most six parcels, three from the larboard side and three from the starboard side. Let $j1$-$j3$ be the larboard and $j4$-$j6$ be the starboard parcels of crude $c1$. Similarly, let $j7$-$j9$ be the larboard and $j10$-$j12$ be the starboard parcels of crude $c2$. Now, a vessel cannot withdraw two parcels at the same time from a given position, this the parcels of any given side cannot transfer simultaneously. Thus, $j1$-$j3$ and $j7$-$j9$ must transfer at different times, if they indeed transfer in the optimal schedule. Since each set consists of three identical and interchangeable parcels, it does not matter which of $j1$-$j3$ (or $j7$-$j9$) transfers first. Therefore, we fix the order of the parcels as $j3$ after $j2$ after $j1$, as $j6$ after $j5$ after $j4$, as $j9$ after $j8$ after $j7$, as $j12$ after $j11$ after $j10$. By fixing these parcel orders, we reduce redundant alternatives for the optimizer without affecting the optimal solution. Note that we cannot preorder the parcels of different crudes, because their parcels are neither identical nor interchangeable, and we have no idea of the sequence in which they may transfer. For example, in Figure 5.1, at position $p1$ (larboard), the first service of crude $c1$ ($j1$) is served before the second service of crude $c2$ ($j8$) but after the first service of the same crude ($j7$).

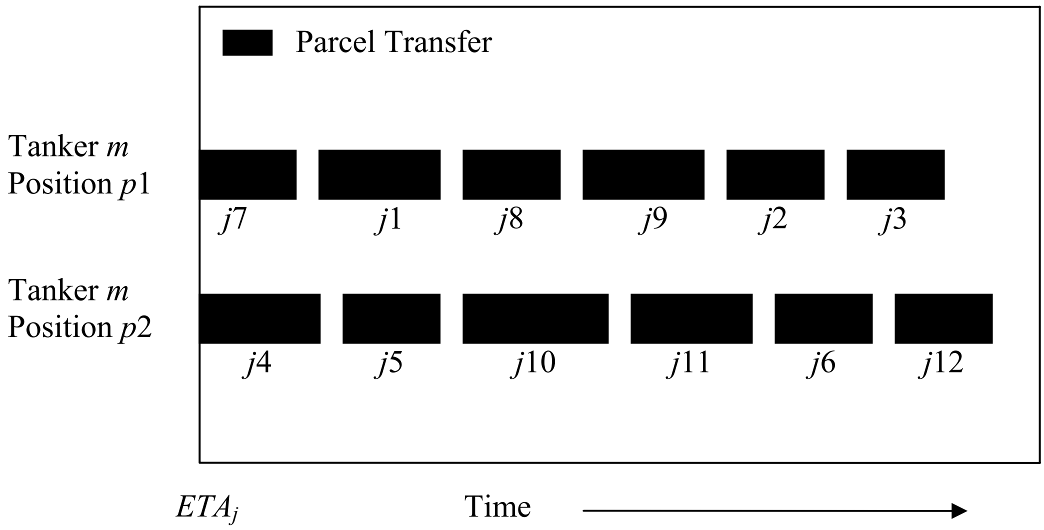

Figure 5.1: Parcel assignments to large tanker m (NS_m^U = 3, $|\boldsymbol{CT}_m|$ = 2)

The above parcel generation procedure will normally result in more parcels than what will actually transfer, as some parcels will not transfer. Let us say that we can postulate an upper limit NS_m^U on the number of distinct parcels that actually transfer from tanker *m*. In many practical situations, a tanker is so small that one or two parcels are sufficient to lighter it, i.e. $NS_m^U = 2$. In this case, the question of two or more parcels transferring simultaneously from any given position will never arise. Thus, we need not assign a transfer position for each parcel, because we know that a tanker can indeed discharge two parcels simultaneously. In this case, each parcel has only two defining attributes, namely the tanker and crude. Therefore, for such small tankers, we define only one parcel for each crude without assigning any position to each parcel and without replicating any parcels. Although in some cases, we may need to distinguish the first service of such a parcel from its second service. This method reduces the number of parcels, hence the number of binary variables and the combinatorial complexity of the problem. To implement this distinction between tankers, we define the set of large tankers (as opposed to small tankers described above) as $\boldsymbol{LT}$ = {*m* | tanker *m* requires more than two parcels for lightering}.

Having defined all the potential parcels fully, we now assign a unique density (ρ_j), an earliest time of arrival (ETA_j), destination refinery, distance to destination refinery (dr_j), distance from anchorage (da_j), parcel transfer rate to vessel *n* (FIN_{nj}), total time (MDT_j) for mounting to and dismounting from its tanker, total time (DTR_j) to dock at and undock from its destination refinery. MDT_j includes the times for connecting and disconnecting hose, setting up transfer equipment, draining/cleaning hose, etc. For two crude parcels withdrawn consecutively by a vessel from the same tanker in a single visit, the vessel must disconnect from the first compartment and then connect to another compartment or keep the same hose connection but change valves. All these operations require certain setup time, so we assume MDT_j to be a parcel-dependent parameter.

Each parcel has a unique crude and a unique tanker. In addition, each parcel of a large tanker ($m \in \mathbf{LT}$) has a unique position. Therefore, we now define the following sets:

$\mathbf{JC}_c$ = {j | parcel j has crude oil c}

$\mathbf{JB}_m$ = {j | parcel j is from tanker m}

$\mathbf{J}_p$ = {j | $m(j) \in \mathbf{LT}$ and parcel j transfers from position p}

where $m(j)$ denotes the tanker to which parcel j belongs. Some crudes may have special transport requirements such as heating. For instance, a heavy crude may be highly viscous and may require heating. A vessel may not be equipped to carry a certain crude. Therefore, we define $\mathbf{JS}_n$ = {j | vessel n can carry parcel j}.

Finally, we assume that vessel does not change its transfer position during a single visit to any tanker. This is because it would be unproductive to do so.

Using these preliminary modeling concepts and notation, we now develop the constraints for our MILP formulation. We first develop the constraints for the operations of vessels. Then, we do the same for tankers. Since the activities on vessels and tankers must match each other, we then write the constraints to couple their operations. Lastly, we define two suitable scheduling objectives. Throughout this chapter, we write each constraint for all valid values of its defining indices, unless stated otherwise.

5.2.1 Lightering Vessels

We divide the time axis of vessel n into K_n contiguous, ordered slots (k = 1, 2, ..., K_n) of unknown, variable lengths as shown in Figure 5.2. A new slot begins, whenever the vessel begins receiving a new parcel. Let TS_{nk} denote the time at which slot k begins on vessel n. We assume that a vessel can receive at most one parcel in each slot. Thus, a vessel n during slot k has two choices. It either remains idle or serves any one of the parcels that it can possibly serve. To model this, we define one binary and one 0-1 continuous variable as follows:

$$x_{nkj} = \begin{cases} 1 & \text{if vessel } n \text{ transfers parcel } j \text{ during slot } k \\ 0 & \text{otherwise} \end{cases} \qquad j \in \boldsymbol{JS}_n,\ 1 \le k \le K_n$$

$$U_{nk} = \begin{cases} 1 & \text{if vessel } n \text{ transfers a parcel during slot } k \\ 0 & \text{otherwise} \end{cases} \qquad 1 \le k \le K_n$$

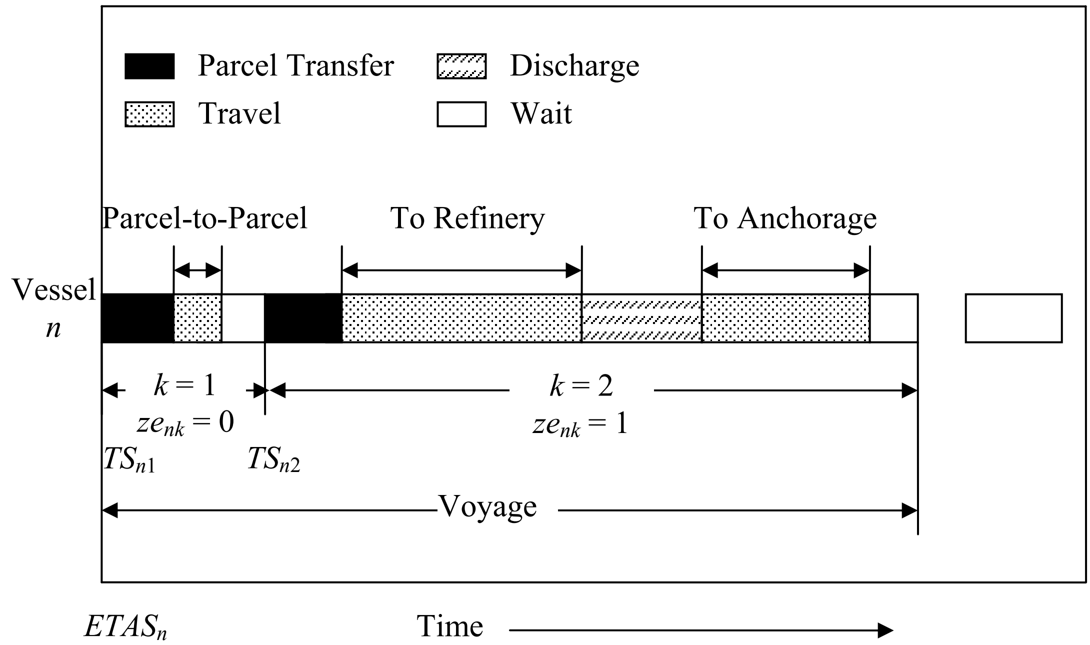

Figure 5.2: Slot-based approach for lightering vessel n

A vessel n can receive at most one parcel during slot k, therefore,

$$U_{nk} = \sum_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (5.1)$$

Because all parcels may not actually transfer and we do know the exact number of parcels that a vessel may actually receive, some empty slots may exist. It is better to push such empty slots to the end of the schedule than leave them in the middle to avoid possible errors in calculating the travel times between parcels. To achieve this, we write,

$$U_{n(k+1)} \le U_{nk} \qquad k < K_n \qquad (5.2)$$

Using some heuristic computations, we can estimate the limits (NS_m^L and NS_m^U) on the number of parcels that a tanker may actually transfer to lighter itself fully. Therefore,

$$NS_m^L \le \sum_n \sum_k \sum_j x_{nkj} \le NS_m^U \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (5.3)$$

We can estimate NS_m^U by rounding up the ratio of LW_m to the smallest capacity (weight) of a vessel. However, if the smallest vessel is far smaller than the others, then this method grossly overestimates the maximum number of parcels and makes the model unnecessarily more difficult. Then, we may use some heuristics to reduce this upper limit. For example, if a single vessel with a large capacity can lighter a tanker, then it is unlikely that the optimal solution will use several small vessels. This is because additional vessels incur additional operating costs. On the other hand, it may also help to use two vessels, as the tanker can offload from two sides simultaneously and lighter itself faster. Any way, using such heuristic rules, we can estimate reasonable NS_m^U and NS_m^L.

5.2.1.1 Voyages

A vessel may have multiple compartments and can carry multiple crudes, thus it is essential that it can pick up several crudes and serve multiple tankers during one voyage during congestion. Because we restricted each slot to at most one parcel, it is clear that a voyage may span multiple slots, if it is to serve multiple tankers and carry multiple crudes. To do this, we need a way to decide the start/end of a voyage. To model this decision, we define the following binary variable:

$$ze_{nk} = \begin{cases} 1 & \text{if vessel } n \text{ ends its current voyage in slot } k \\ 0 & \text{otherwise} \end{cases} \qquad 1 \le k \le K_n$$

If vessel n is not on a voyage during slot k, then it cannot end its voyage, so,

$$ze_{nk} \le U_{nk} \tag{5.4}$$

Clearly, a vessel n can do two things at the end of slot k. It may continue its current voyage and move to another parcel in the next slot, or it may terminate its voyage and travel to its destination refinery to deliver the crudes. To distinguish between these two scenarios, we define a 0-1 continuous variable:

$$Z_{nkjj'} = \begin{cases} 1 & \text{if vessel } n \text{ transfers parcel } j \text{ in slot } k,\ j' \text{ in } (k+1) \text{ during the same voyage} \\ 0 & \text{otherwise} \end{cases}$$

$$j \,\&\, j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n$$

where, $\boldsymbol{JJ} = \{(j, j') \mid \text{some vessel can carry parcels } j \text{ and } j' \text{ during the same voyage}\}$. Recall that each voyage has a single destination refinery, so a vessel in one voyage can carry only the parcels with the same destination refinery. However, we exclude from $\boldsymbol{JJ}$ the parcel pairs with the same crude and tanker, as we do not want a vessel to withdraw the same crude multiple times during any voyage.

From the definition, $Z_{nkjj'} = x_{nkj}x_{n(k+1)j'}(1-ze_{nk})$. If vessel n does not receive parcel j during slot k (i.e. $x_{nkj} = 0$), then it cannot travel from parcel j to any other parcel in slot (k+1). Therefore,

$$x_{nkj} \ge \sum_{j'} Z_{nkjj'} \qquad j \,\&\, j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n \tag{5.5a}$$

Similarly, it cannot travel to parcel j' in slot (k+1), if it does not receive parcel j' (i.e. $x_{n(k+1)j'} = 0$). Thus,

$$x_{n(k+1)j'} \ge \sum_{j} Z_{nkjj'} \qquad j \,\&\, j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n \tag{5.5b}$$

If vessel n terminates a voyage at slot k (i.e. $ze_{nk} = 1$), then it obviously cannot go from any parcel to any other parcel directly, that is,

$$1 - ze_{nk} \geq \sum_{j} \sum_{j'} Z_{nkjj'} \qquad j \;\&\; j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n \qquad (5.5c)$$

Lastly, $Z_{nkjj'} = 1$, only when vessel n takes parcel j in slot k, j' in $(k+1)$, and continues its current voyage, therefore,

$$Z_{nkjj'} \geq x_{nkj} + x_{n(k+1)j'} - ze_{nk} - 1 \qquad j \;\&\; j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n \qquad (5.5d)$$

Eqs. 5.5a-d are exact linear equivalents for the non-linear definition of $Z_{nkjj'}$ and ensure that it is binary.

Lastly, the number of voyages (i.e. the number of last slots of voyages) for a lightering vessel n in the planning horizon must equal the number of slots with parcels minus the number of parcel-to-parcel transitions within the same voyage. In other words,

$$\sum_{k} ze_{nk} = \sum_{k} U_{nk} - \sum_{k<K_n} \sum_{j} \sum_{j'} Z_{nkjj'} \qquad j \;\&\; j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ} \qquad (5.6)$$

<u>5.2.1.2 Parcel Transfer</u>

Recall that we defined unique parcels for large tankers, so such a parcel cannot transfer more than once:

$$\sum_{n} \sum_{k} x_{nkj} \leq 1 \qquad j \in \boldsymbol{JS}_n,\ j \ni m(j) \in \boldsymbol{LT} \qquad (5.7)$$

We do not need the above for parcels from small tankers, as we allow the same parcel to transfer twice as described earlier.

As a vessel moves from parcel to parcel during a voyage, it receives parcels into its various compartments. Even though we prefixed the identity of each parcel, we still need to decide the amount of crude in it and the number of vessel compartments required for its storage. Let VP_{nkj} ($j \in \boldsymbol{JS}_n$) be the volume of parcel j

that vessel n withdraws during slot k and NC_{nk} be the number of compartments that store parcel received by vessel n during slot k. Clearly, the vessel cannot carry more than its physical capacity, cannot violate the draft limitation of its destination refinery, and cannot withdraw more than what the tanker has, so

$$VP_{nkj} \leq x_{nkj} \min[N_n^U SC_n, \frac{WD_{nj}}{\rho_j}, PS_c] \qquad j \in \boldsymbol{JS}_n \qquad (5.8a)$$

where, c is the crude in parcel j and WD_{nj} is the maximum weight of parcel j that vessel n can carry. Note that WD_{nj} depends on the allowable draft for vessel n at the destination refinery of parcel j. Eq. 5.8a also ensures that parcel volume is zero, when the parcel does not transfer.

When a tanker carries a crude whose amount is less than or equal to the lightering amount, i.e. $PS_c \leq LW_m/\rho_c$, then it is possible that vessels withdraw all of this crude during lightering operations. Eq. 8a assures that a single parcel does not withdraw more crude than what the tanker has. However, multiple vessels may withdraw a given crude or one vessel may withdraw a given crude multiple times in different voyages, therefore we must make sure that the total volume of a crude withdrawn through all parcels does not exceed the available volume. Therefore, for such crudes, we enforce,

$$\sum_n \sum_k \sum_j VP_{nkj} \leq PS_c \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JC}_c, c \ni PS_c \leq LW_m/\rho_c \qquad (5.8b)$$

where, crude c belongs to tanker m.

Recall that a vessel may have multiple compartments and it must store each crude in a different compartment. In addition to limiting the number of crudes that the vessel can carry, this will also limit the crude load, because the vessel may not use all compartments fully for its crudes. Furthermore, to ensure that a vessel n does not use more than N_n^U compartments, we must assign vessel compartments to each crude. The minimum number of compartments used by vessel n during slot k is given by,

$$\left(NC_{nk} - 1\right)SC_n \le \sum_j VP_{nkj} \le NC_{nk}SC_n \qquad j \in \boldsymbol{JS}_n \qquad (5.9)$$

This constraint also ensures that the volume of crude in each compartment does not exceed the compartment size.

Of course, the parcel sizes should be sufficient to lighter each tanker adequately. In other words,

$$\sum_n \sum_k \sum_j VP_{nkj}\rho_j = LW_m \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (5.10)$$

Recall that we allow a vessel to carry multiple crudes during a voyage. As a vessel moves from one parcel to another during a voyage, it accumulates crudes. Let WT_{nk} (VT_{nk}) denote the total weight (volume) of crudes collected, and NT_{nk} denote the total number of compartments used by vessel n up to and including slot k on the current voyage or trip. We can compute WT_{nk}, VT_{nk}, and NT_{nk} respectively by using the following constraints,

$$WT_{n(k+1)} = WT_{nk}(1 - ze_{nk}) + \sum_j \rho_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.11a)$$

$$WT_{n1} = \sum_j \rho_j VP_{n1j} \qquad j \in \boldsymbol{JS}_n \qquad (5.11b)$$

$$VT_{n(k+1)} = VT_{nk}(1 - ze_{nk}) + \sum_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.12a)$$

$$VT_{n1} = \sum_j VP_{n1j} \qquad j \in \boldsymbol{JS}_n \qquad (5.12b)$$

$$NT_{n(k+1)} = NT_{nk}(1 - ze_{nk}) + NC_{n(k+1)} \qquad k < K_n \qquad (5.13a)$$

$$NT_{n1} = NC_{n1} \qquad (5.13b)$$

Eqs. 5.11a, 5.12a, and 5.13a are nonlinear and of similar forms. Therefore, we can use the same method to linearize them. Let us first linearize eqs. 5.11a.

There are two ways of linearizing eq. 5.11a. We can either define a new variable $WZ_{nk} = WT_{nk}(1 - ze_{nk})$ and linearize it or use the following constraints:

$$WT_{n(k+1)} \geq WT_{nk} + \sum_j \rho_j VP_{n(k+1)j} - WT_n^U ze_{nk} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.14a)$$

$$WT_{n(k+1)} \leq WT_{nk} + \sum_j \rho_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.14b)$$

$$WT_{n(k+1)} \leq \sum_j \rho_j VP_{n(k+1)j} + WT_n^U (1 - ze_{nk}) \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.14c)$$

$$WT_{nk} \geq \sum_j \rho_j VP_{njk} \qquad k > 1, j \in \boldsymbol{JS}_n \qquad (5.14d)$$

$$WT_n^U = \max_{j \in \boldsymbol{JS}_n} WD_{nj}$$

Eqs. 5.15a-b are critical for all slots of a voyage, while eq. 5.15c is critical for the first only.

The alternate approach is to substitute $WZ_{nk} = WT_{nk}(1 - ze_{nk})$ in eq. 5.11a and then linearize it to get,

$$WT_{n(k+1)} = WZ_{nk} + \sum_j \rho_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.15a)$$

$$WZ_{nk} \leq WT_{nk} \qquad (5.15b)$$

$$WZ_{nk} \leq WT_n^U \left(1 - ze_{nk}\right) \qquad (5.15c)$$

$$WZ_{nk} + WT_n^U ze_{nk} \geq WT_{nk} \qquad (5.15d)$$

Using the above procedure, we can get the following for eq. 5.12a.

$$VT_{n(k+1)} \geq VT_{nk} + \sum_j VP_{n(k+1)j} - VT_n^U ze_{nk} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.16a)$$

$$VT_{n(k+1)} \leq VT_{nk} + \sum_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.16b)$$

$$VT_{n(k+1)} \le \sum_j VP_{n(k+1)j} + VT_n^U (1 - ze_{nk}) \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.16c)$$

$$VT_{nk} \ge \sum_j VP_{njk} \qquad k > 1, j \in \boldsymbol{JS}_n \qquad (5.16d)$$

$$VT_n^U = \max_{j \in \boldsymbol{JS}_n} \frac{WD_{nj}}{\rho_j}$$

$$VT_{n(k+1)} = VZ_{nk} + \sum_j \rho_j VP_{n(k+1)j} \qquad k < K_n, j \in \boldsymbol{JS}_n \qquad (5.17a)$$

$$VZ_{nk} \le VT_{nk} \qquad (5.17b)$$

$$VZ_{nk} \le VT_n^U \left(1 - ze_{nk}\right) \qquad (5.17c)$$

$$VZ_{nk} + VT_n^U ze_{nk} \ge VT_{nk} \qquad (5.17d)$$

where, $VZ_{nk} = VT_{nk} (1 - ze_{nk})$.

Similarly, for eq. 5.13a, we obtain,

$$NT_{n(k+1)} \ge NT_{nk} + NC_{n(k+1)} - N_n^U ze_{nk} \qquad k < K_n \qquad (5.18a)$$

$$NT_{n(k+1)} \le NT_{nk} + NC_{n(k+1)} \qquad k < K_n \qquad (5.18b)$$

$$NT_{n(k+1)} \le NC_{n(k+1)} + N_n^U (1 - ze_{nk}) \qquad k < K_n \qquad (5.18c)$$

$$NT_{nk} \ge NC_{nk} \qquad k > 1 \qquad (5.18d)$$

$$NT_{n(k+1)} = NZ_{nk} + NC_{n(k+1)} \qquad k < K_n \qquad (5.19a)$$

$$NZ_{nk} \le NT_{nk} \qquad (5.19b)$$

$$NZ_{nk} \le N_n^U \left(1 - ze_{nk}\right) \qquad (5.19c)$$

$$NZ_{nk} + N_n^U ze_{nk} \ge NT_{nk} \qquad (5.19d)$$

with $NZ_{nk} = NT_{nk} (1 - ze_{nk})$.

The first linearization method involves eqs. 5.11b, 5.12b, 5.13b, 5.14, 5.16, and 5.18; the second involves eqs. 5.11b, 5.12b, 5.13b, 5.15, 5.17, and 5.18. At the first glance, the second alternative seems to have the same number of constraints, but with three additional continuous variables. However, it has two equalities that define the cumulative variables (eqs. 5.11b and 5.14a for WT_{nk}, eqs. 5.12b and 5.17a for VT_{nk}, and eqs. 5.13b and 5.19a for NT_{nk}), which we can eliminate. Therefore, we eliminate these variables and equalities completely from the formulation by substituting for WT_{nk}, VT_{nk}, and NT_{nk} using those equalities. For the sake of brevity, we do not write the resulting constraints here. However, we do eliminate them, when we solve examples later. As a result, the second linearization method requires the same number of continuous variables but fewer constraints than the first. However, we cannot say which linearization method is better without doing a detailed numerical evaluation as done later.

Having computed the cumulative crude weight, compartments, and volume (WT_{nk}, NT_{nk}, and VT_{nk}) for a vessel n during slot k, we now impose the draft and compartment limits on these variables.

First, the total weight WT_{nk} on vessel n must not exceed the maximum weight that the vessel can carry to its destination refinery due to draft limitations. As stated earlier, the parcels transferred during a single voyage have a common destination. From the parcel transferred during each slot, we know the destination refinery for that voyage. Therefore,

$$WT_{nk} \leq \sum_{j} WD_{nj} x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (5.20)$$

Second, the total number of compartments used during any slot should not exceed N_n^U. Therefore, we demand,

$$NT_{nk} \leq N_n^u U_{nk} \qquad (5.21)$$

Previously, eq. 5.9 ensured that the volume of crude in each compartment does not exceed the compartment size. Now, eqs. 5.9 and 5.21 together guarantee that the load in vessel *n* will never exceed the vessel's physical volume of $SC_n N_n^U$.

5.2.1.3 Operation Timings

So far, we have allocated physical assets such as vessels, compartments, etc. to various operations, but have not addressed the actual timing of each operation. To this end, we define the various travel times required to service parcel *j* under different circumstances:

$TPR_{nj} =$ Time for vessel *n* to take parcel *j* to its destination refinery $= dr_j / vf_n$

$TRA_{nj} =$ Time for vessel *n* to travel from destination refinery of parcel *j* to anchorage $= \left(dr_j - da_j\right) / ve_n$

$TAP_{nj} =$ Time for vessel *n* to travel from anchorge to pick up parcel $j = da_j / ve_n$

$TPP_{njj'}$ = Time for vessel *n* to travel from parcel *j* to *j′* directly at speed vf_n

As described earlier, a voyage begins with an empty vessel from anchorage receiving its first parcel and ends with the vessel returning to the next parcel via the destination refinery. Clearly, the operations that the vessel undergoes in a slot vary with where the slot is in a voyage.

If a slot is not the last in a voyage, then the vessel travels from the current parcel to the next and waits for the next service to begin as shown in Figure 5.2. In other words,

$$TS_{n(k+1)} \geq TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + \sum_j \sum_{j'} TPP_{njj'} Z_{nkjj'}$$

$$j \,\&\, j' \in \boldsymbol{JS}_n,\ (j, j') \in \boldsymbol{JJ},\ k < K_n \qquad (5.22a)$$

Recall that TS_{nk} is the time at which vessel *n* begins receiving a parcel during slot *k*. The second term on the right involves the times for mounting/dismounting and then transferring crude oil. The third term is the travel time from the current parcel to the next, as the voyage continues. Although meant for slots other than the last in a voyage, eq. 5.22a holds even for the last slot, as $Z_{nkjj'} = 0$ for the last slot and the other operations take place in the last slot anyway.

During the last slot of a voyage, the vessel receives a parcel, travels to the refinery to discharge all crudes, and returns to the anchorage again as shown in Figure 5.2. In other words,

$$TS_{n(k+1)} \geq TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + \sum_j TPR_{nj} x_{nkj} + \left(\sum_j DTR_j x_{nkj} + \frac{VT_{nk}}{FOUT_n} \right)$$
$$+ \sum_j \left(TRA_{nj} x_{nkj} + TAP_{nj} x_{n(k+1)j} \right) - HT\left(1 - ze_{nk}\right)$$

$$j \in \boldsymbol{JS}_n,\ k < K_n \qquad (5.22b)$$

where, big-M constant HT is greater than the planning horizon. The terms on the right includes starting time of slot k, service time of parcel j, travel time to the refinery, discharge time at the refinery and travel time to the next lightering location (anchorage/offshore) respectively. This constraint is activated only when slot k is the last slot in a voyage (i.e. $ze_{nk} = 1$).

Now, we know the tanker can start lightering operation only after both tanker and vessel arrive at the lightering location (anchorage/offshore). We write the following two constraints for tankers and lightering vessels respectively,

$$TS_{nk} \geq \sum_j ETA_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (5.23)$$

$$TS_{n1} \geq ETAS_n + \sum_j TAP_{nj} x_{n1j} \qquad j \in \boldsymbol{JS}_n \qquad (5.24)$$

The second term on the right of eq. 5.24 ensures that a vessel n receives its first parcel, only after it arrives at the first parcel's transfer position, which could be either anchorage or offshore.

Also note that, for offshore tankers, they travel to the anchorage for second lightering service after the first lightering service completes at the offshore. Therefore, the start time of second lightering service of a tanker is dependent on its departure time from offshore. It is the sum of the departure time of offshore tanker m (DTB_m) from the offshore location and the travel time of this tanker (TOA_m) from offshore to anchorage location. We use the following constraint to model this,

$$TS_{nk} \geq DTB_m + TOA_m - HT\left(1 - \sum_j x_{nkj}\right)$$

$$(m, m'(j)) \in \boldsymbol{MM}, j \in \boldsymbol{JS}_n \cap \boldsymbol{S2} \cap \boldsymbol{JB}_{m'} \qquad (5.25)$$

where, set $\boldsymbol{S2}$ = { j | stage two (anchorage) parcels of 2-stage tankers}. The above constraint activates only when vessel n serves a stage two parcel from the corresponding anchorage tanker m' of the offshore tanker m in slot k, where both m and m' are the same 2-stage tanker. Finally, we impose a lower bound to TS_{nk}, which is $ETAS_n$.

5.2.1.4 Arrival Times at Refineries

The arrival time of crudes at a refinery is an important indicator of the customer satisfaction level. It is better to deliver the crudes to the destination refinery as planned without any delay. Recall that we have two types of parcels, one for small tankers that can transfer at most twice; while the other for large tankers that can transfer at most once. We define the arrival times of these two types of parcels differently.

Let ATR_{nkj} denote the arrival time of a parcel j of a small tanker at its destination refinery via vessel n during slot k, and ATR_j denote the arrival time of a parcel j of a large tanker at the refinery. For the crude transferred during the last slot, the arrival time equals the start time of this slot plus the transfer time and travel time to the refinery. Hence, we write,

$$ATR_{nkj} \geq TS_{nk} + \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + TPR_{nj} x_{nkj} - HT\left(2 - ze_{nk} - x_{nkj}\right)$$

$$j \in \boldsymbol{JS}_n, j \ni m(j) \notin \boldsymbol{LT} \qquad (5.26a)$$

$$ATR_j \geq TS_{nk} + \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + TPR_{nj} x_{nkj} - HT\left(2 - ze_{nk} - x_{nkj}\right)$$

$$j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.26b)$$

The above constraints activate, only when parcel j transfers during slot k to vessel n (i.e. $x_{nkj} = 1$) and this slot is the last in a voyage (i.e. $ze_{nk} = 1$).

If any two parcels transfer consecutively in a voyage, then the arrival time of the former is the same as the arrival time of the latter. We guarantee this by imposing,

$$ATR_{nkj} \geq ATR_{n(k+1)j'} - HT\left(1 - Z_{nkjj'}\right)$$

$$k < K_n, j \,\&\, j' \in \boldsymbol{JS}_n, (j, j') \in \boldsymbol{JJ}, j \ni m(j) \notin \boldsymbol{LT}, j' \ni m'(j') \notin \boldsymbol{LT} \qquad (5.27a)$$

$$ATR_{nkj} \geq ATR_{j'} - HT\left(1 - Z_{nkjj'}\right)$$

$$k < K_n, j \,\&\, j' \in \boldsymbol{JS}_n, (j, j') \in \boldsymbol{JJ}, j \ni m(j) \notin \boldsymbol{LT}, j' \ni m'(j') \in \boldsymbol{LT} \qquad (5.27b)$$

$$ATR_j \geq ATR_{n(k+1)j'} - HT\left(1 - Z_{nkjj'}\right)$$

$$k < K_n, j \,\&\, j' \in \boldsymbol{JS}_n, (j, j') \in \boldsymbol{JJ}, , j \ni m(j) \in \boldsymbol{LT}, j' \ni m'(j') \notin \boldsymbol{LT} \qquad (5.27c)$$

$$ATR_j \geq ATR_{j'} - HT\left(1 - \sum_n \sum_k Z_{nkjj'}\right)$$

$$k < K_n, j \,\&\, j' \in \boldsymbol{JS}_n, (j, j') \in \boldsymbol{JJ}, j \ni m(j) \in \boldsymbol{LT}, j' \ni m'(j') \in \boldsymbol{LT} \qquad (5.27d)$$

The above four constraints include all possible combinations of parcels, whether from large tankers or small tankers.

Lastly, we impose a lower bound on ATR_{nkj} (ATR_j), which is the due date of parcel j (DD_j) for $j \in \boldsymbol{JS}_n$.

5.2.2 Tankers

The operations in tankers are the same as the big ships in the general transshipment problem addresses in Chapters 3-4 except for the calculation of service time. Hence, we will use the same methodology, do the necessary modifications and present the constraints in the following section. Note that, we only concern about the operations on large tankers ($m \in \boldsymbol{LT}$). Therefore, in this section, we write the constraints and define the variables only for large tankers.

Let T_j denote the start time of unloading parcel j and define binary variable $y_{jj'}$ as follows to model their partial sequence of parcel transfers.

$$y_{jj'} = \begin{cases} 1 & \text{if parcel } j' \text{ transfers later than } j \\ 0 & \text{otherwise} \end{cases} \qquad j \;\&\; j' \in \boldsymbol{J}_p \text{ for some } p$$

As described earlier, we fix the sequence of parcels that originate from the same crude and unload from the same position. Therefore, $y_{jj'} = 1$ for $p(j) = p(j')$, $c(j) = c(j')$, $j < j'$, $j \ni m(j) \in \boldsymbol{LT}$. Alternatively, we can also set $y_{jj'}$ to 0 without affecting the solution. However, consistency has to be maintained throughout the problem.

We ensure at most one service any time for a given position by forcing one parcel can start transfer only after the previous parcel has completed its lightering from the same position. Therefore,

$$T_{j'} \geq T_j + \sum_n \sum_k \left(\frac{VP_{nkj}}{FIN_{nj}} + MDT_j x_{nkj} \right) - HT\left(1 - y_{jj'}\right)$$

$$p(j) = p(j'), j < j', j \ni m(j) \in \boldsymbol{LT}, j \in \boldsymbol{JS}_n, c(j) \neq c(j') \qquad (5.28a)$$

$$T_j \geq T_{j'} + \sum_n \sum_k \left(\frac{VP_{nkj'}}{FIN_{nj'}} + MDT_{j'} x_{nkj'} \right) - HT y_{jj'}$$

$$p(j) = p(j'), j < j', j \ni m(j) \in \boldsymbol{LT}, j' \in \boldsymbol{JS}_n, c(j) \neq c(j') \qquad (5.28b)$$

where, $p(j)$ is $p \ni j \in \boldsymbol{J}_p$. The above constraints only sequence parcel pairs that are from different crudes of a tanker.

For parcels that are from the same crude, we have prefixed the sequence of them ($y_{jj'} = 1$). Therefore, eq. 5.28a is always activated and calculates the slot length, while eq. 5.28b is never activated. We write,

$$T_{j'} \geq T_j + \sum_n \sum_k \left(\frac{VP_{nkj}}{FIN_{nj}} + MDT_j x_{nkj} \right)$$

$$p(j) = p(j'), j < j', j \ni m(j) \in \boldsymbol{LT}, j \in \boldsymbol{JS}_n, c(j) = c(j') \qquad (5.28c)$$

A parcel can start offloading only when the vessel n receiving it has arrived at the lightering location (anchorage/offshore), so we have,

$$T_j \geq \sum_n \sum_k \left(ETAS_n + TAP_{nj} \right) x_{nkj} \qquad j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.29a)$$

Some parcels are not served. We then push them to the front of the schedule by forcing such parcels to start at the earliest possible time.

$$T_j \leq ETA_j + HT \sum_n \sum_k x_{nkj} \qquad j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.29b)$$

In addition, we impose lower bound, $T_j \geq ETA_j$ for $j \ni m(j) \in \boldsymbol{LT}$.

Lastly, the start times of a common parcel in both tanker and vessel are matched by using the following two coupling constraints,

$$T_j \geq TS_{nk} - HT \left(1 - x_{nkj} \right) \qquad j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.30a)$$

$$T_j \leq TS_{nk} + HT \left(1 - x_{nkj} \right) \qquad j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.30b)$$

5.2.3 Additional constraints

In Chapter 3, we showed that including additional constraints that fix some transfer sequences and sequences of unserved parcels on the large tankers improved the model performance in a general transshipment problem. Therefore, we also develop additional constraints for our model.

We push the empty parcels to the beginning of the schedule by using,

$$y_{jj'} + \delta_{jj'} = \sum_{n'} \sum_{k} x_{n'kj'} - \sum_{n} \sum_{k} x_{nkj} + 1$$

$$p(j) = p(j'), j \in \boldsymbol{JS}_n, j' \in \boldsymbol{JS}_{n'}, j < j', j \ni m(j) \in \boldsymbol{LT} \quad (5.31)$$

where, $\delta_{jj'}$ is a dummy 0-1 continuous variable.

Then, we fix those transfer sequences that are obvious from other transfer sequences by using,

$$y_{jj''} + d_{jj'j''} = y_{jj'} + y_{j'j''} \qquad p(j) = p(j') = p(j''), j < j' < j'', j \ni m(j) \in \boldsymbol{LT} \quad (5.32)$$

Where, $d_{jj'j''}$ is a dummy 0-1 continuous variable.

5.2.4 Objective

To calculate the system cost, we need to determine the departure times of tankers, the total distance and voyages traveled by vessel n during the entire planning horizon, and the arrival time of crude oil at the refinery port. We have calculated the last term in section 5.2.1.4. Now, we will develop equations to calculate the remaining variables in the following section.

5.2.4.1 Tanker Departure Time

A tanker can depart only when required amount of crude has been unloaded. That means, the departure time of a tanker is greater than the start time of any slot receiving parcel from it plus the service time of parcel in that slot,

$$DTB_m \geq TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) - HT \left(1 - \sum_j x_{nkj} \right) \quad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \quad (5.33a)$$

In addition, for large tankers ($m \in \boldsymbol{LT}$), the tanker's departure time is greater than the start time of any parcel associated with the tanker plus the lightering operation time of that parcel. We write,

$$DTB_m \geq T_j + \sum_n \sum_k \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) \qquad j \in \boldsymbol{JS}_n, j \ni m(j) \in \boldsymbol{LT} \qquad (5.33b)$$

The lower bound of departure time is $DTB_m \geq ETAB_m + TOA_m$, where TOA_m is 0 for 1-stage tanker.

Unlike general transshipment problem, we do not need the departure time of vessel n. Lightering vessels travel multiple trips between refineries and lightering locations. Hence, we are more interested at the distances and voyages traveled by the lightering vessels.

5.2.4.2 Vessel Traveling Distance

The total traveling distance of lightering vessel (TD_n) is the sum of the round trip distance between all the parcels received and their destination refineries, minus the round trip distance of parcels that are picked up in the same voyage, and plus the traveling distance from one parcel to another in the same voyage. In other words,

$$TD_n = \sum_j \sum_k 2dr_j x_{nkj} - \sum_j \sum_{j'} \sum_{k<K_n} \left(2dr_j - TPP_{njj'} vf_n \right) Z_{nkjj'}$$

$$j \,\&\, j' \in \boldsymbol{JS}_n, (j, j') \in \boldsymbol{JJ} \qquad (5.34)$$

5.2.4.3 Objective Function 1

The refineries pay time-charter cost to the tanker owners even though these tankers sit idle for lightering services or waiting for services. Therefore, one important objective is to minimize the time-charter cost of crude tankers. The total system cost (TC) is then the sum of due date penalty of crudes, fuel cost, fixed operating cost of lightering vessels and time-charter cost of tankers. It is written as follows,

$$TC = \sum_{n} \sum_{\substack{j \in \boldsymbol{JS}_n \& \\ j \ni m(j) \notin \boldsymbol{LT}}} \sum_{k=1}^{K_n} DDP_j \left(ATR_{nkj} - DD_j \right) + \sum_{j \ni m(j) \in \boldsymbol{LT}} DDP_j \left(ATR_j - DD_j \right)$$

$$+ \sum_{n} TD_n FC_n + \sum_{n} \sum_{k=1}^{K_n} VC_n ze_{nk} + \sum_{m \notin \boldsymbol{TST}} TCC_m \left(DTB_m - ETAB_m \right) \tag{5.35a}$$

$$+ \sum_{(m,m') \in \boldsymbol{MM}} TCC_m \left(DTB_{m'} - ETAB_m - TOA_m \right)$$

where, DDP_j is the daily due date penalty of parcel j if it is delivered to the refinery by the vessel later than the agreed due date (DD_j). In addition, FC_n is the fuel cost of vessel n per mile, VC_n is the fixed operating cost of vessel n per voyage, and TCC_m is the time-charter cost of tanker m. The first two terms calculate the due date penalty of crudes that are delivered later than the due date, for crudes from small and large tankers respectively. However, if a crude is delivered on time or earlier than due date, ATR_{nkj} (ATR_j) takes the lower bound DD_j; hence, the due date penalty is 0. The third term is the fuel cost of vessels. The fourth term is the fixed operating cost of vessels. The fifth and sixth terms are the time-charter costs of 1-stage tankers and 2-stage tankers respectively. Note that, for a 2-stage tanker, the traveling from offshore to anchorage is a continuation of its journey, which does not affect the lightering schedule. Thus, we do not include it in calculating the time spent for lightering service for 2-stage tankers. The lightering time then equals the departure time from the anchorage minus the arrival time at the offshore, and minus the travel time from offshore to anchorage.

5.2.4.4 Objective Function 2

Sometimes, a lightering company contracts with the refineries to finish lightering crude tankers within an agreed duration (AD_m). If the tankers spend more time for lightering than the agreed duration, the lightering company has to pay the refineries demurrage charges. To model this, we modify our objective function and lower bound slightly as follows,

$$TC = \sum_{n} \sum_{\substack{j \in \boldsymbol{JS}_n \& \\ j \ni m(j) \notin \boldsymbol{LT}}} \sum_{k=1}^{K_n} DDP_j \left(ATR_{nkj} - DD_j \right) + \sum_{j \ni m(j) \in \boldsymbol{LT}} DDP_j \left(ATR_j - DD_j \right)$$

$$+ \sum_{n} TD_n FC_n + \sum_{n} \sum_{k=1}^{K_n} VC_n ze_{nk} + \sum_{m \notin \boldsymbol{TST}} DC_m \left(DTB_m - ETAB_m - AD_m \right) \tag{5.35b}$$

$$+ \sum_{(m,m') \in \boldsymbol{MM}} DC_m \left(DTB_{m'} - ETAB_m - TOA_m - AD_m \right)$$

where, DC_m is the demurrage cost of tanker *m*. Demurrage is only paid for the extra time required in addition to the agreed duration. Hence, we subtract the agreed duration from the lightering time to obtain the extra time. And the rest of the terms remain unchanged.

The lower bound of departure time of tanker (DTB_m) is thus modified to,

$$DTB_m \geq ETAB_m + TOA_m + AD_m \tag{5.36}$$

With this lower bound, if a tanker *m* leaves the anchorage earlier than the agreed duration, the demurrage cost is forced to 0.

Now, we complete our two alternate tanker lightering models (M1 And M2). Both M1 and M2 includes eqs. 5.1-5.10, 5.11b, 5.12b, 5.13b, and 5.20-5.32. In addition, M1 uses eqs. 5.14, 5.16 and 5.18; while M2 uses eqs. 5.15, 5.17 and 5.19. And we use two different objectives, one considers time-charter cost of tankers (eq. 5.35a); the other considers demurrage cost of tankers (eq. 5.35b and lower bound Eq. 5.36).

In the next chapter, we evaluate these models with different objectives in detail using several examples.

CHAPTER 6

TANKER LIGHTERING OPERATIONS – MODEL EVALUATION

Having developed two alternate MILP formulations and identified two suitable objectives to address the tanker lightering problem, we now use several examples to evaluate their important features and compare the performance of proposed models. In this chapter, we first use three examples to investigate the impact of different objectives on the tanker lightering schedules, as well as to compare the performance of different formulations. In addition, Examples 1 and 2 aim to study the applications and highlight the important features of models. Example 3 focuses on the effective solution of a large problem by means of several heuristic methods. Lastly, we compare our simplified slot-based model with an event-based model (Lin et al., 2003) to evaluate the performances of these two types of formulations.

6.1 Examples

For a fair and consistent comparison, we solve all the examples in this chapter, including those of Lin's models (Lin et al., 2003), using CPLEX 9.0 with GAMS 21.7 on a PC with Intel Pentium IV 3.2 GHz CPU and 2.0 GB of RAM running Windows XP (SP2). The relative termination tolerance is set to 0.0%. The big-M (*HT*) values used are the same for Examples 1-3, which is 400.

We use Gantt charts to present the tanker lightering schedules for Examples 1-3. Four types of rectangles with different patterns are used to represent different types of activities. The black rectangle represents the parcel transfer (inclusive of setup/mounting/dismounting), the dotted rectangle represents the vessel travel (parcel-to-parcel, parcel-to-refinery, refinery-to-anchorage, etc.), the slashed rectangle represents the discharge operation at the refinery (inclusive of docking/undocking), and the white rectangle represents the waiting for the next parcel transfer. We use one line to represent one lightering vessel. We tag the tankers lightered and refineries visited by a lightering vessel on top of the respective slots. The first number inside each black rectangle denotes the weight of crude transferred during that slot ($\sum_j \rho_j VP_{nkj}$), the number inside the bracket denotes the number of compartments used to store the crude received in that slot (NC_{nk}). Similarly, inside each slashed rectangle, the first number represents the total weight of crudes discharged in that refinery (WT_{nk}), and the number inside the bracket represents the cumulative compartments used in that voyage (NT_{nk}). We also label the starting time of a lightering service (TS_{nk}) beneath the corresponding slot. Lastly, we use one time line (same time axis) to denote the departure time of each tanker at the bottom of the chart, where tankers are labeled on top of the line, and its corresponding departure time (DTB_m) is shown beneath the line.

6.1.1 Example 1

We first consider a simple system with $M = 5$ tankers carrying $C = 6$ crudes and $N = 3$ lightering vessels. The data for this example are shown in Tables 6.1-4. The first tanker (*m*1) is a large one-crude tanker that requires at least 3 parcel transfers (NS^L_{m1} = 3 and NS^U_{m1} = 3). Therefore, we assign six possible parcels (*j*1- *j*6) to it, 3 at starboard, 3 at larboard. The second tanker (*m*2) is a small one-crude tanker ($NS^U_{m2} \le 2$), so we assign only one parcel (*j*7) to it. The third tanker (*m*3) is a two-crude tanker (*c*3 and *c*4). However, the lightering weight is small ($NS^U_{m3} \le 2$), so two parcels (*j*8 and *j*9) are needed, one for each crude. The fourth (*m*4) and fifth (*m*5) one-crude tankers are actually the same 2-stage tanker where *m*4 is the offshore tanker, and *m*5 is the corresponding anchorage tanker. The lightering weights at the two stages are small ($NS_{m4}^U \leq 2$ and $NS_{m5}^U \leq 2$)). Hence, we use one parcel for each tanker respectively ($j10$ and $j11$). In addition, it takes tanker $m4$ 6 hours to travel from offshore to anchorage. Among all the crudes, crude $c2$ from tanker $m2$ and crude $c4$ from tanker $m3$ have common destination refineries $r2$. Therefore, set ***JJ*** includes parcel pairs ($j7, j9$) and ($j9, j7$), and the travel time betweens them are 3 hours, same for all the vessels.

Table 6.1: Data for tankers in Examples 1-3

Examples 1-2					
Tanker m	$ETAB_m$ (h)	LW_m (ton)	TCC_m (\$/h)	DC_m (\$/h)	AD_m (h)
1	0.0	638	1200	1300	15.0
2	5.0	320	750	1000	30.0
3	5.0	170	1000	1000	30.0
4	90.0	250	900	800	100.0
5	90.0	200	900	800	100.0
Example 3					
1	0.0	198	700	700	100.0
2	5.0	600	1400	1400	100.0
3	20.0	170	1000	1000	100.0
4	22.0	210	1300	1300	100.0
5	23.0	177	900	900	100.0
6	62.0	400	1000	1000	100.0
7	62.0	200	1000	1000	100.0

Table 6.2: Data for crudes in Examples 1-3

Crude c	Tanker m	Refinery r	Heating	PS_c (m^3)	ρ_c (ton/m^3)	MDT_c (h)	DTR_c (h)	DDP_c (\$/h)	DD_c (h)	dr_c (nm)	da_c (nm)
Examples 1-2											
1	1	1	No	9000	0.85	3.0	3.0	500	30.0	88	-
2	2	2	No	3000	0.85	3.0	3.0	500	100.0	60	-
3	3	3	No	6000	0.85	3.0	3.0	500	100.0	70	-
4	3	2	No	6000	0.90	3.0	3.0	500	100.0	60	-
5	4	4	Yes	3000	0.85	3.0	3.0	500	400.0	120	40
6	5	4	Yes	8000	0.85	3.0	3.0	500	400.0	80	-
Example 3											
1	1	1	Yes	6800	0.85	3.0	3.0	500	30.0	88	-
2	2	2	Yes	6000	0.85	3.0	3.0	500	50.0	60	-
3	3	3	No	8550	0.90	3.0	3.0	500	50.0	80	-
4	3	1	No	6000	0.85	3.0	3.0	500	50.0	88	-
5	4	3	No	6000	0.90	3.0	3.0	500	50.0	80	-
6	5	1	No	2000	0.85	3.0	3.0	500	50.0	88	-
7	6	4	Yes	6000	0.85	2.5	3.0	600	40.0	110	40
8	7	4	Yes	6500	0.85	2.5	3.0	600	40.0	70	-

nm = nautical mile

Table 6.3: Data for lightering vessels in Examples 1-3

Vessel n	Heating	$ETAS_n$ (h)	$FOUT_n$ (m^3/h)	vf_n (nm/h)	ve_n (nm/h)	N_n^u	SC_n (m^3)	VC_n (\$/voyage)	FC_n (\$/nm)
Examples 1-2									
1	Yes	0.0	20	8.0	11.0	4	100	6200	20
2	No	0.0	15	7.0	9.5	3	100	6800	20
3	Yes	0.0	20	9.0	12.0	2	130	8200	20
Example 3									
1	Yes	0.0	20	8.0	11.0	4	100	6200	20
2	Yes	0.0	15	7.0	9.5	3	110	6800	20
3	Yes	0.0	20	9.0	12.0	2	150	8200	20
4	Yes	0.0	15	7.0	9.0	2	120	9000	20

Table 6.4: Data for loading capacities and pumping rates of lightering vessels in Examples 1-3

Examples 1-2												
Vessel	Loading Capacity WD_{nr} (ton)				Pumping Rate FIN_{nc} (m^3/h)							
n	*r*1	*r*2	*r*3	*r*4	*c*1	*c*2	*c*3	*c*4	*c*5	*c*6		
1	315	315	315	250	60	60	60	60	60	60		
2	190	190	190	210	38	38	38	38	38	38		
3	220	220	220	210	50	50	50	50	50	50		
Example 3												
n	*r*1	*r*2	*r*3	*r*4	*c*1	*c*2	*c*3	*c*4	*c*5	*c*6	*c*7	*c*8
1	330	330	330	280	60	60	60	60	60	60	60	60
2	300	300	300	250	55	55	55	55	55	55	55	55
3	250	250	250	210	50	50	50	50	50	50	50	50
4	200	200	200	200	38	38	38	38	38	38	38	38

We first implement the data using the objective that considers time-charter cost (TCC) of tankers (Example 1a) with both models (M1 and M2). Then, we implement the same set of data using the objective for demurrage cost (DC) of tankers (Example 1b) with both models (M1 and M2) as well. We use K = 4 slots for this example. Table 6.5 summarizes the computational results and model statistics. The problem is of medium size with 148 discrete variables (binary and integer variables). The optimizer requires twelve to fifty minutes to generate the solution. Both M1 and M2 have the same rMIP and MIP objective values in both examples. However, M2 seems to perform better than M1. But it is rash to jump to the conclusion that M2 is better than M1 with only one example. Therefore, we will compare their performances using more examples later on.

Table 6.5: Model statistics and computational results for Examples 1- 3

	Example 1a (TCC)		Example 1b (DC)	
	M1	M2	M1	M2
Binary Variables	148	148	148	148
Continuous Variables	289	289	289	289
Constraints	1036	1027	1036	1027
Non zeros	5615	5426	5615	5426
Nodes	561076	404043	279561	125560
rMIP objective ($)	68131.30	68131.30	55733.80	55733.80
MIP objective ($)	191172.84	191172.84	116378.12	116378.12
Integrality gap (%)	64.36	64.36	52.11	52.11
CPU time (s)	2637	1902	1245	738
	Example 2a (TCC)		Example 2b (DC)	
Binary Variables	148	148	148	148
Continuous Variables	289	289	289	289
Constraints	1036	1027	1036	1027
Non zeros	5615	5426	5615	5426
Nodes	398775	336958	280748	75707
rMIP objective ($)	68131.30	68131.25	55733.80	55733.80
MIP objective ($)	210537.87	210537.87	144011.27	144011.27
Integrality gap (%)	67.64	67.64	61.30	61.30
CPU time (s)	1945	1697	1511	437
	Example 3a (TCC)		Example 3b (DC)	
Binary Variables	131	131	131	131
Continuous Variables	340	340	340	340
Constraints	1090	1090	1090	1090
Non zeros	5863	5740	5863	5740
Nodes	1061793	1176362	86844	80296
rMIP objective ($)	86395.29	86395.29	73960.00	73960.00
MIP objective ($)	319019.99	319019.99	265342.61	265342.61
Integrality gap (%)	72.92	72.92	72.13	72.13
CPU time (s)	4848	5418	437	342

Figures 6.1-2 show the Gantt charts for tanker lightering schedules with different objectives respectively. Let us first look at the schedule that minimizes the total time-charter cost of tankers (Example 1a, Figure 6.1). Tankers $m1$, $m2$, and $m3$ arrive at the anchorage within a short time such that they compete for the three lightering vessels. The time-charter cost for $m1$ is higher than $m2$ and $m3$ (1200 $/h vs.

750 \$/h vs. 1000 \$/h). Thus, $m1$ has highest priority and it starts offloading first among the three.

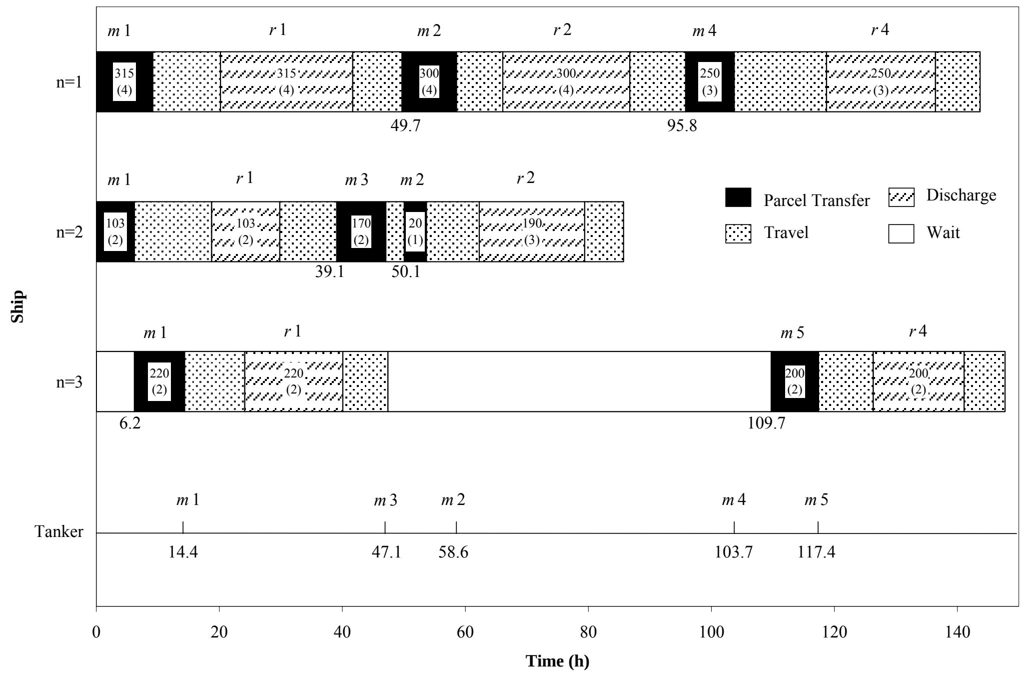

Figure 6.1: Tanker lightering schedule for Example 1a (TCC)

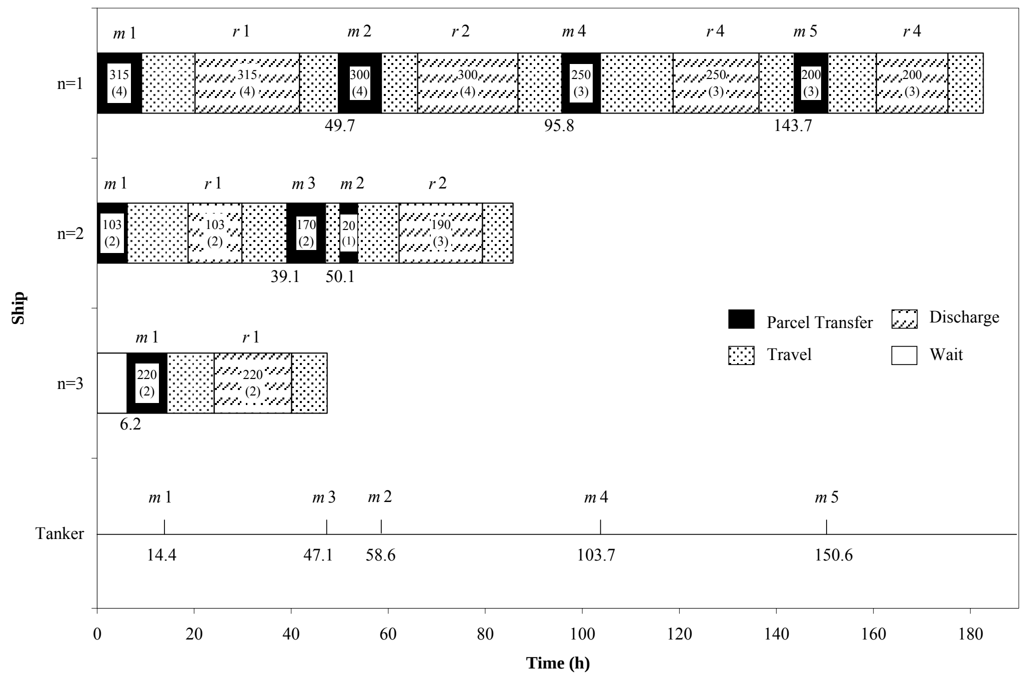

Figure 6.2: Tanker lightering schedule for Example 1b (DC)

In addition, *m*1 is a large tanker that requires three services. As described in Chapter 5, at most two services can take place at a tanker simultaneously. The resultant lightering schedule shows that our model guarantees this, where *n*3 starts offloading *m*1 (the third service) only after *n*2 finishes lightering *m*1 at the same side.

Tanker *m*3 is a two-crude tanker that carries both crude *c*3 and *c*4 with different densities and destinations. The optimal schedule chose to offload *c*4. That is because it is heavier (0.90 ton/m^3 vs. 0.85 ton/m^3), hence less volume (transfer time) is required for meeting the lightering requirement. Furthermore, the destination of *c*4 is nearer than *c*3 (60 nm vs. 70 nm), so lightering vessel travels a shorter distance, spends less travel time, and consumes less fuel to deliver *c*4. Note that, in this case, the optimizer is free to decide which crude to offload. However, in some cases, the lightering weight of each crude is specified according to the client refineries' requirements. We can model this simply by setting the crude size to its specific lightering weight though the actual crude size may be larger.

Recall that, crude *c*2 (*m*2) and *c*4 (*m*3) have the same destination refinery *r*2. Therefore, it is possible for a lightering vessel to load them consecutively in a voyage to save round-trip travel time. In fact, Gantt chart (Figure 6.1) shows that *n*2 picks up *c*4 first, then picks up *c*2 at the anchorage before departing for refinery *r*2. By such practice, it reduces one voyage, travel time and fuel cost of one-round trip travel for lightering vessels.

Tankers *m*4 and *m*5 are actually one physical 2-stage tanker, where *m*4 represents the tanker at offshore and *m*5 represents the same tanker at anchorage. Our models included this consideration. As shown in the schedule, tanker *m*5 starts its service at 109.7 h only after *m*4 departs from offshore (103.7 h) and travels to the anchorage (6 h).

Among the three lightering vessels ($n1$, $n2$ and $n3$) that the company operates, $n1$ is the most efficient and economical one, since it has the least fixed voyage cost (VC_{n1} = 6200 \$/voyage), the largest size ($N_{n1}^{U} SC_{n1}$ = 400 m^3), fast transfer rates (FIN_{n1c} = 60 m^3/h and $FOUT_{n1}$ = 20 m^3/h) and velocities (ve_{n1} = 11.0 nm/h and vf_{n1} = 8.0 nm/h) among the three. Therefore, it is not surprising to see that its utilization is the highest. It travels three voyages, while the rest only travel two voyages. In addition, its total transfer amount is the largest (765 ton).

Now, let us investigate the impact on the lightering schedule when the objective is to minimize the demurrage of tankers. In this example, we deliberately set the agreed duration (AD_m) of tankers $m4$ and $m5$ to 100 h such that both of them can spend a long time at the lightering location without increasing the total cost. The lightering schedule (Figure 6.2) shows that $m5$ is now lightered by the most efficient and economical vessel $n1$ instead of $n3$ in the previous example. However, its departure time from anchorage increases from 117.4 h to 150.6 h.

The DC schedule is less compact than the TCC schedule. For minimizing demurrage, as long as the tanker spends less than agreed duration for lightering operation, the lightering company need not pay extra money. Therefore, some flexibility occurs since time is not limiting, if the agreed duration is longer than the actual operation time. Thus, a tanker now can wait at the lightering location until the most efficient lightering vessel becomes free to service it. On the other hand, minimizing time-charter cost aims to minimize the total time tanker spent idling. Thus, the schedule is normally very compact such that the departure time for each tanker is minimized. The solution shows that tradeoff exists between utilization of vessels and demurrage of tankers.

6.1.2 Example 2

The previous example showed that receiving multiple crudes with a common destination from multiple tankers in a voyage is cost effective since it reduces the operating cost of lightering vessels and the demurrage cost of the tankers. However, it delays the delivery time of the first crude in a voyage. Therefore, it is not always economical to aggregate crudes with a common refinery. To elaborate this, in this example, we set the due date requirement of crudes from tanker *m*3 to 0 h. Other parameters are the same as previous example.

Table 6.5 also lists the model statistics and computational results for this example. Again, M2 requires shorter time to solve the problem than M1. Figures 6.3-4 show the lightering schedules for TCC and DC respectively. Both schedules show that *n*2 does not lighter *m*3 and *m*2 consecutively in the same voyage anymore. Once *n*2 finishes lightering *c*4 from *m*3, it then travels directly to *r*2 to ensure the crude can reach the refinery port as early as possible. As a result, the total number of voyages increases to 8; whereas, in Example 1, the total number of voyages is only 7. Although this increases the operating cost of vessels, it ensures a fast delivery of *c*4 to refinery *r*2.

When we compare schedules 2a (TCC) and 2b (DC), schedule 2a is again much more compact than 2b. In addition, the utilization of *n*1 is very high considering the demurrage of tankers. It travels four voyages and lighters 296 ton of crude from *m*2. On the other hand, when we consider the time-charter cost of tankers, the utilization of *n*1 is low. It only travels three voyages and lighters 130 ton from *m*2. However, *m*2, *m*4, and *m*5 depart much earlier in Example 2a than Example 2b (55.3 h vs. 58.5 h for *m*2; 97.9 h vs. 103.4 h for *m*4; 111.6 h vs. 150.3 h for *m*5).

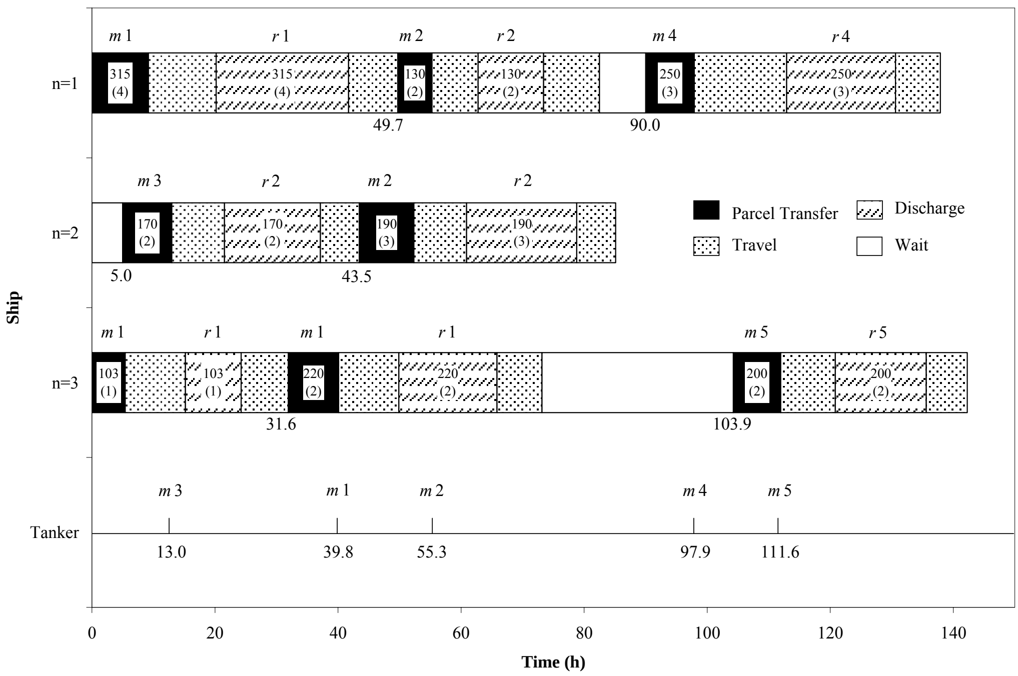

Figure 6.3: Tanker lightering schedule for Example 2a (TCC)

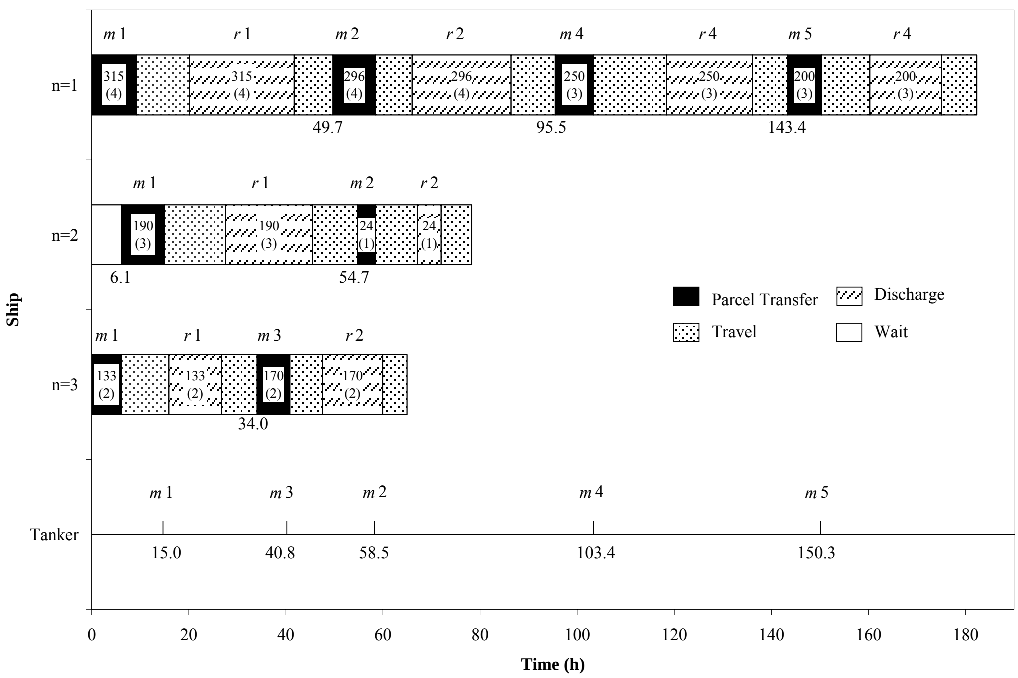

Figure 6.4: Tanker lightering schedule for Example 2b (DC)

6.1.3 Example 3 (Heuristics)

So far, we have presented some small examples to illustrate the application and features of our models. Because of the practical features we consider, our formulation is complex. Hence, it requires long solution time to generate optimal solutions for real practical problems with hundreds to thousands of discrete variables. However, in most cases, a good (not necessarily optimal) solution with reasonable solution time is sufficient for the company to enhance the efficiency of lightering operations and save large sums of money. To this end, we develop some heuristic methods to simplify the problem based on common knowledge and practical experience in this section.

First, we have already used one heuristic method when postulating the maximum number of distinct parcels that a large tanker may transfers (NS_m^U) as described in Chapter 5.

Second, our models decide on which crude to lighter for a tanker with multiple crudes. Generally, two properties of a crude affect the selection, namely, density and destination refinery. The density of a crude affects the actual lightering volume hence the crude transfer time. For a given lightering weight, a heavier crude results in less volume and shorter pumping time. Therefore, it is normally optimal to lighter the heavier crude first. Furthermore, a crude with shorter delivery distance (distance of its designated refinery) is normally favorable in minimizing the operating cost. That is because the lightering vessel then travels shorter distance and needs less time. This reduces its fuel consumption cost, and waiting time of the subsequent tankers for its service. Therefore, for a tanker carrying multiple crudes, we need not generate possible parcels for a light crude with long delivery distance. By this method, we can reduce number of parcels and hence the problem size considerably.

Third, previously, a lightering vessel is allowed to pick up multiple parcels of the same destination in a voyage. We include all those parcel pairs from different crudes with common designated refinery when defining set ***JJ***. However, if a parcel arrives early, and another parcel with a common destination refinery arrives much later, it is unlikely for a lightering vessel to pick up the early parcel first then wait for sometime to pick up the latter parcel. Such operation not only increases the delivery time of the first parcel, it also reduces the utilization of the lightering vessel since it spends time idling. It is unlikely optimal in this case. Therefore, we will only include those parcel pairs with a common destination and reasonably close arrival times for set ***JJ***. Furthermore, if some crude is needed urgently by the refinery (i.e. a very high due date penalty or a short due date), then it is better to deliver it as soon as possible. Hence, such crudes are unlikely to transfer as the first crude in a voyage. But it is possible to be the last crude of a voyage, as this may save waiting time for suitable vessels.

Lastly, tankers arrive at irregular time intervals. Some arrive very early, while some arrive later. For those early tankers, lightering vessels can finish servicing them using the first few slots. In other words, the early tankers do not appear in the last few slots of a lightering vessel. However, it is still possible to serve the later tankers by first few slots as well as last few slots of lightering vessels. We can exclude the later slots for early tankers by fixing the corresponding x variables to 0. By this method, we reduce some unnecessary slot-tanker combinations. Therefore, we reduce the size and complexity of the problem.

With all the above heuristic methods excluding some unlikely optimal combinations, we now can handle more realistic and complex tanker lightering problems.

Let us use an example to illustrate the above heuristic methods. The problem size is large with $M = 7$ tankers, $C = 8$ crudes and $N = 4$ lightering vessels. Tables 6.1-4 give the detailed information. In addition, $m6$ and $m7$ are the respective offshore and anchorage tankers of the same 2-stage tanker. The travel time from offshore to anchorage for this tanker is 5 h.

We first assign a reasonable number of parcels to each tanker. Except for *m*2, all the tankers are small tankers that require no more than two services. In addition, multi-crude tanker *m*3 has two types of crudes *c*3 and *c*4. *c*3 is heavier and has shorter delivery distance than *c*4 (0.90 ton/m^3 vs. 0.85 ton/m^3 and 80 nm vs. 88 nm). Its size is greater than the lightering weight of the tanker. Therefore, we assume only *c*3 transfers. As a result, we assign one parcel to each small tanker, one parcel to *c*3 of tanker *m*3, and six parcels to large tanker *m*2. We have twelve parcels in total.

We use three slots (K = 3) for this problem. Tankers *m*1 and *m*2 arrive very early (0.0 h and 5.0 h), therefore, they are likely to finish lightering in the first two slots. That is, $x_{n3j} = 0$ for $j \ni m(j) = m1$ and *m*2.

Among all the lightering crudes, both *c*1 and *c*6 deliver to *r*1, both *c*3 and *c*5 deliver to *r*3. However, *c*1 and *c*6 arrive at the anchorage at long time intervals (0 h and 23 h). Thus, we only include parcel pairs for *c*3 and *c*5 in set ***JJ***. The travel time between them is 3 h for all the lightering vessels.

With such simplifying methods, the models require only 131 discrete variables (Table 6.5) even though the problem is large. The optimizer finds the solution within 5500 seconds. Figures 6.5-6 show the schedules for minimal time-charter cost and demurrage cost of tankers respectively.

On the other hand, without any heuristic methods (except for the postulation of NS_m^U), the models require 171 discrete variables. Furthermore, none of them converges to optimum even after 6500 s.

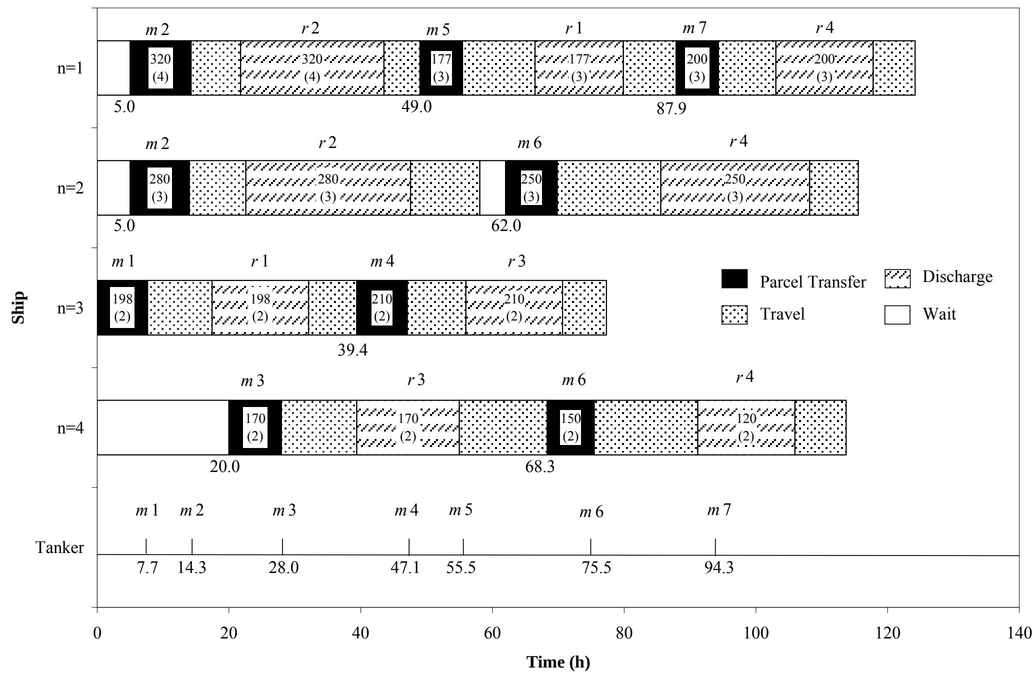

Figure 6.5: Tanker lightering schedule for Example 3a (TCC)

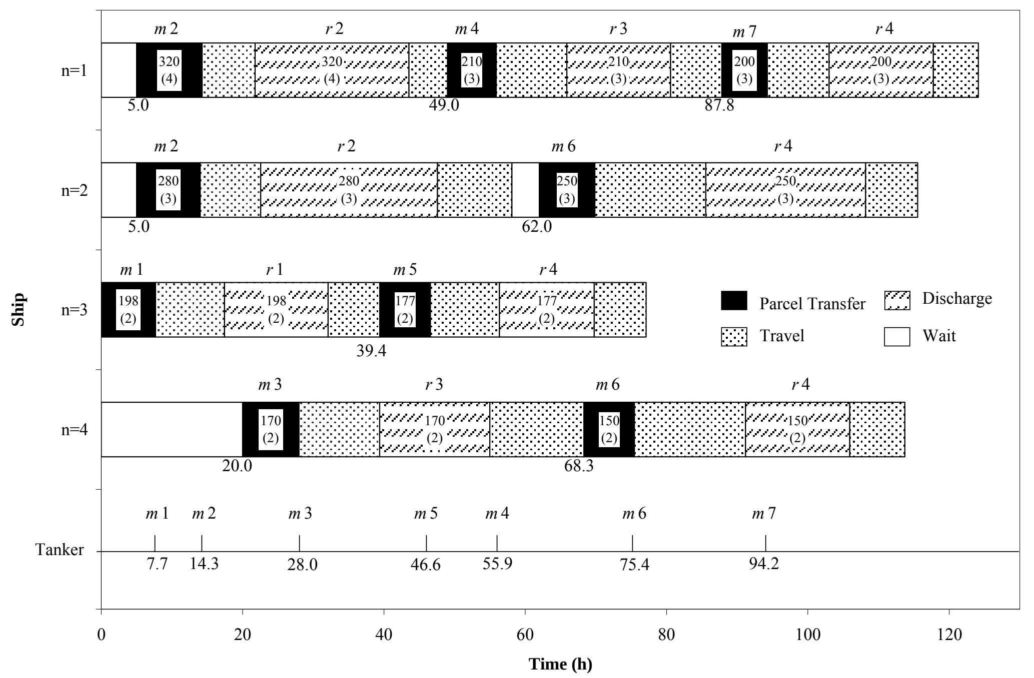

Figure 6.6: Tanker lightering schedule for Example 3b (DC)

6.1.4 Remarks

Having presented the results and statistics for all the examples, we now study the model performance in a systematic way. First, we compare the different linearization methods. In all the examples, both M1 and M2 have the same rMIP values. Thus, neither of them is superior to the other in terms of formulation tightness. However, M2 always requires fewer nonzeros and sometimes fewer constraints (5426 vs. 5615 in Examples 1-2) than M1. This smaller size contributes to the effectiveness of M2, where M2 outperforms M1 in most examples (2637 s vs. 1902 s in Example 1a; 1245 s vs. 738 s in Example 1b; 1945 s vs. 1697 s in Example 2a; 1511 s vs. 437 s in Example 2b; 437 s vs. 342 s in Example 2b) except for Example 3a.

The second observation is related to the model performance with different objectives. Though the model statistics are identical with different objectives, the results show that minimizing TCC is much harder than minimizing DC. For example, in Examples 3a-b, TCC problem requires over 5000 s, while DC problem requires less than 500 s. TCC problem normally results in more compact schedules, and the time constraint is normally tighter. It is therefore harder to find out the optimal solution. In addition, the integrality gap for TCC problem is always greater than that for DC problem (64.36% vs. 52.11% in Example 1; 67.64% vs. 61.30% in Example 2; 72.92% vs. 72.13% in Example 3), which also indicates that it is harder to solve TCC problem than DC problem.

6.2 Comparison with Event-based Formulation

Previously, Lin et al. (2003) addressed a tanker lightering problem using event-based formulation in a limited form without considering some practical features as described earlier. However, we are interested in studying the model performances of both event-based and slot-based approaches. For a fair comparison, we first implement Cases 1-4 from their work using their reported formulations. Then, we use the same set of example data and implement our slot-based formulation in a reduced form by assuming the following,

(1) All crude tankers within the time horizon are small tankers such that at most two transfers are sufficient to lighter fully. Therefore, no constraints are needed to model the operations of tankers.

(2) The lightering vessels have only one compartment.

(3) Only one visit to one tanker per voyage is allowed for a lightering vessel.

(4) For a tanker carrying multiple crudes, the selection of which crude and in what amount to lighter is known and fixed a priori.

(5) Densities of different crudes are uniform.

(6) Empty slots are allowed in the middle of the schedule with zero length.

(7) The total cost is only dependent on the time-charter cost of tanker and the fixed operating costs of vessels.

(8) In Cases 1-3, all the tankers are 1-stage tankers that only lighter at anchorage.

With these assumptions, we remove all the constraints and variables addressing the above features. Note that the alternate linearization methods are not included since the cumulative variables are not needed. The objective in minimizing the time-charter cost of tankers is considered. Furthermore, the removal of constraints for crude tankers reduces all the sequenced-based variables and constraints. As a result, we have two simple formulations, one for Cases 1-3, the other for Case 4; and both of them use only the slot-based approach for lightering vessels with variable slot-lengths. For convenience, we name the former O1, the latter O2. Similarly, we called Lin's respective models as L1 and L2. The detailed formulations of O1 and O2 are included in appendices C.1 and C.2.

The example data are taken from Lin et al. (2003). For full details, please refer Lin et al. 2003. They used a heuristic method to simplify the formulations by defining a subset of event points for each tanker. However, these subsets are not reported in their work. We select our own subsets carefully such that the objective values and tanker lightering schedules are the same as the reported results. Table 6.6 gives the subsets defined for Cases 1-4. In addition, the big-M values are 168, 520, 528 and 240 respectively for Cases 1, 2, 3, and 4.

Applying the same heuristic method, our formulations are able to solve all the examples and obtain the same tanker lightering schedules with the same total costs as Lin's models. Table 6.7 compares the model statistics and computational results for both models. The binary variables used in Lin's models and our models are the same. However, our slot-based models have far fewer continuous variables, constraints, and non-zeros than Lin's event-based models. For example, in Case 1, L1 requires 202 continuous variables, 619 constraints and 2016 nonzeros; whereas, our model O1 requires only 92 (half) continuous variables, 200 (one third) constraints and 864 (half) nonzeros. The model statistics for Cases 2-4 also show the same observations.

Table 6.6: Subset data for Section 6.2

Tanker Number	Event Point (Slot Number) *			
	Case 1	Case 2	Case 3	Case4
1	1, 2	1	1	1, 2
2	1, 2	1	1	1, 2
3	1, 2	2	1	1, 2
4	2, 3	2	2	2, 3
5	2, 3	2, 3	1, 2	2, 3
5′	2, 3	-	-	2, 3
6	2, 3	3, 4	1, 2	2, 3
7	3	4, 5	1, 2	3
8	-	-	2, 3	-
8a	-	-	-	3, 4
8b	-	4, 5	-	3, 4
9	-	4, 5	2, 3	-
10	-	4, 5	2, 3	-
11	-	5, 6	3	-
12	-	6	3	-

* L's models use event point; O's models use slot number

Table 6.7: Model statistics and computational results for Section 6.2

	Case 1		Case 2		Case 3		Case 4	
	L1	O1	L1	O1	L1	O1	L2	O2
Binary Variables	60	60	70	70	114	114	74	74
Continuous Variables	202	92	223	131	362	169	235	116
Constraints	619	200	892	463	1234	459	880	440
Non zeros	2016	864	3033	1455	4157	1804	2961	1360
Nodes	11877	8008	24153	13955	9968486	5353325	7158	1650
rMIP objective ($)	36.64	48.50	39.28	84.94	52.88	72.79	55.85	73.78
MIP objective ($)	190.40	190.40	173.69	173.69	159.58	159.58	310.21	310.21
Integrality gap (%)	80.76	74.53	77.39	51.10	66.86	54.38	82.00	76.22
CPU time (s)	9.2	5.7	18.5	11.9	14161.8	6639.0	7.9	1.9

In event-based formulations, normally two time variables are defined to denote the start and end of an event respectively. On the other hand, our formulations use only one time variable to represent the start of a slot. Furthermore, event-based models normally require lots of big-M constraints for ordering different start/end time of different events (Sundaramoorthy and Karimi, 2005). However, our formulations generally order the slots without using big-M constraints. Because of these fundamental differences, our slot-based formulations generally require fewer variables and constraints, and thus are simpler as compared with Lin's models.

Therefore, we expect our models outperform Lin's models. From the results, it is not surprising to observe that our models indeed are tighter with higher rMIP values and perform better with faster CPU times; Case 1 ($48.50 vs. $36.64 and 5.7s vs. 9.2s), Case 2 ($84.94 vs. $39.28 and 11.9s vs. 18.5s), Case 3 ($72.79 vs. $52.88 and 6639s vs. 14162s) and Case 4 ($73.78 vs. $55.85 and 1.9s vs. 7.9s), where the former value is from our models and the latter is from Lin's models.

6.3 Conclusion

In this chapter, we used several examples to show the important features of our formulations, such as, the selection of type of crude and the amount of crude to lighter per visit, the number and sequence of tankers to service per voyage, handling of 2-stage lightering operation, and so on.

Generally, the schedule that considers tanker demurrage is more flexible than the schedule that minimizes the time-chart cost of tankers. It is thus easier to solve the DC problem with shorter CPU times. In addition, the utilization of lightering vessels in the DC problem is normally higher than that in the TCC problem, because the tanker cost is generally lower in the DC problem.

Both two alternate linearization methods (M1 and M2) are capable of finding optimal solutions. However, M2 is simpler with fewer constraints and nonzeros. As a result, M2 performs better than M1 in most cases.

Furthermore, we have developed some heuristic methods to reduce slot-tanker combinations, thus the problem size. By doing this, the model is capable of obtaining a good solution within a reasonable time for large size problem. We considered a large size problem with seven tankers, eight crudes and four lightering vessel. Without heuristic methods, the models require 171 discrete variables and do not converge after 6500 s. However, with heuristic methods, the models require only 131 discrete variables (23.40% size reduction) and converge to optimum within 5500 s.

Lastly, the comparison of model performances between slot-based and event-based formulations shows that slot-based formulation is generally simpler and more effective. It is smaller in size and has tighter formulation, thus is faster than the event-based formulation.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

In this research, we addressed two important scheduling problems, general chemical transshipment operation and tanker lightering operation.

We developed nine alternative MILP formulations all using both slot-based and pair-wise sequencing approaches but with different linearization methods to solve general transshipment operations. The research showed that the models with pair-wise sequencing approach for the recipient carriers outperform the models with slot-based approach for the recipient carriers, when the transshipment system involves many recipient carriers. The reverse is true, when the problem have many donor carriers or two-sided cargos. Furthermore, the comparison between different linearization methods shows that the method involving the big-M constraints seem to be the fastest in spite of its inferior rMIP values. As a result, out of the nine alternate formulations, the one with big-M constraints and uses slot-based approach for recipient carriers performs best in most examples. In addition, we developed an effective heuristic method by aggregating cargos with common origins and destinations into one cargo. The results indicate that such method is promising even for practical industrial problems with large sizes.

In the second part of the work, we addressed a special case of chemical transshipment operation – tanker lightering. We developed two alternate formulations considering two different objectives. Here, we showed that the objective that considers the time-charter cost of tankers is generally harder to solve than the objective that considers demurrage cost of tankers. It is because that the optimal schedule for minimal time-charter cost is normally very compact. The departure times of tankers are early but the utilization of vessels is small. In addition, we also showed that the linearization method using variables that combine the cumulative quantity and voyage information is superior to the other one. In addition, we proposed several heuristic methods to simplify the problem for large size problem. Lastly, a comparison study of slot-based and event-based models showed that slot-based model is more efficient than event-based models for tanker lightering problems.

7.2 Recommendations

Both problems deal with direct ship-to-ship chemical transfer at a transfer port or seawater. However, as described earlier, our formulations neglect the port limitations. Hence, one can extend our formulations to consider the physical limits and practical operations at the transfer port, such as, the infrastructure limitations, the number of berths, berths assignments to the ships, and so on.

In addition, for general chemical transshipment problems, the models can incorporate many more practical features. For example, some industrial restrictions on cargo transfer sequence can be considered. Detailed cargo allocation in the multi-compartment recipient carrier is another possible research area of interest.

For tanker lightering problem, we have considered many important practical features. However, one can still address more practical considerations. For example, in practice, it is also possible for a lightering vessel to visit multiple refineries to discharge same/different crude from same/different tankers in a voyage. In addition, the model can be extended to include the effect of tides on the lightering weight, fuel consumption and hence the optimal lightering schedule. Furthermore, one may consider non-identical compartment sizes of lightering vessels.

REFERENCES

(1) Andrews, S., Murphy, F. H., Wang, X. P. and Welch, S., 1996. Modeling crude oil lightering in Delaware bay. *Interfaces*, November-December, 26 (6), 68-78.

(2) Bausch, D. O., Brown, G. G. and Ronen, D., 1998. Scheduling short-term marine transport of bulk products. *Maritime Policy & Management*, 25 (4), 335-348.

(3) Brooke, A., Kendrick, D., Meeraus, A. and Raman, R., 2003. *GAMS: A User's Guide*. Washington DC: GAMS Development Corporation.

(4) Brown, G. G., Graves, G. W. and Ronen, D., 1987. Scheduling ocean transportation of crude oil. *Management Science*, 33 (3), 335-346.

(5) Chajakis, E. D., 1997. Sophisticated crude transportation. *OR/MS Today*, 24 (Dec), 30-34.

(6) Chajakis, E. D., 2000. Management science for marine petroleum logistics. *In:* S. H. Zanakis, G. Doukidis and C. Zopounidis, eds. *Decision making: Recent Developments and Applications*. Dordrecht, Netherlands: Kluwer Academic Publishers, 279-292.

(7) Cheng, L. and Duran, M., 2003. *In:* I. E. Grossman and C. McDonald, eds. *World-wide crude transportation logistics: a decision support system based on simulation and optimization, in Proceedings of Foundations of Computer-Aided Process Operations (FOCAPO 2003): a view to the future integration of R&D, manufacturing and the global supply chain*. Austin, TX: CACHE Corporation.

(8) Christiansen, M., Fagerholt, K., Nygreen, B. and Ronen, D., 2003. Maritime transportation. Working paper, Section of Managerial Economics and

Operations research, Norwegian University of Science and Technology, Trodheim, Norway.

(9) Christiansen, M., Fagerholt, K. and Ronen, D., 2004. Ship routing and scheduling: status and perspectives. *Transportation Science*, 38 (1), 1-18.

(10) Dantzig, G. B. and Fulkerson, D. R., 1954. Minimizing the number of tankers to meet a fixed schedule. *Naval Research Logistics Quarterly*, 1, 217-222.

(11) Daskin, M. S. and Walton, C. M., 1983. An approximate analytical model for supertanker lightering operations. *Transportation Research*, 178 (3), 201-219.

(12) Fagerholt, K. and Christiansen, M., 2000a. A combined ship scheduling and allocation problem. *Journal of Operational Research Society*, 51, 834-842.

(13) Fagerholt, K. and Christiansen, M., 2000b. A traveling salesman problem with allocation, time window and precedence constraints - an application to ship scheduling. *International Transactions in Operational Research*, 7, 231-244.

(14) Fagerholt, K. and Lindstad, H., 2000. Optimal policies for maintaining a supply service in the Norwegian Sea. *OMEGA*, 28, 269-275.

(15) Fisher, M. L. and Rosenwein, M. B., 1989. An interactive optimization system for bulk-cargo ship scheduling. *Naval Research Logistics*, 36, 27-42.

(16) Gupta, S. and Karimi, I. A., 2003. An improved MILP formulation for scheduling multiproduct, multistage batch plants. *Ind. Eng. Chem. Res.*, 42 (11), 2365-2380.

(17) Huang, C. and Karimi, I. A., 2005. Scheduling transshipment operations in maritime chemical transportation. Submitted to *Ind. Eng. Chem. Res*.

(18) Hwang, S. J., Al-Khayyal, F., Ahmed S. and Reveliotis, S., 2002. Ship routing and scheduling for multi-commodities with inventory constrained time windows. Technical report. The Logistics Institute, Georgia Tech, and the Logistics Institute- Asia Pacific, National University of Singapore.

(19) Jetlund, A. S., 2002. *Dynamic routing and scheduling of combination carriers*. Thesis (M.Sci.). National University of Singapore.

(20) Jetlund, A. S. and Karimi, I. A., 2004. Improving the logistics of multi-compartment chemical tankers. *Computers & Chemical Engineering*, 28 (8), 1267-1283.

(21) Karimi, I. A., Srinivasan, R. and Por, L.H., 2002. Unlock supply chain improvements through effective logistics. *Chemical Engineering Progress*, 98 (5), 32-38.

(22) Karimi, I. A., Tan, Z. Y. L. and Bhushan, S., 2004. An improved formulation for scheduling an automated wet-etch station. *Computers & Chemical Engineering*, 29, 217-224.

(23) Lawrence, S. A., 1972. *International sea transport: the years ahead.* Lexington, MA: Lexington Books.

(24) Lin, X., Chajakis, E. D. and Floudas, C. A., 2003. Scheduling of tanker lightering via a novel continuous-time optimization framework. *Ind. Eng. Chem. Res.*, 42, 4441-4451.

(25) Mehrez, A., Hung, M. S. and Ahn, B. H., 1995. An industrial ocean-cargo shipping problem. *Decision Sciences*, 26 (3), 669-677.

(26) MPA, Singapore, 2004. Port Statistics [online]. Available from: http://www.mpa.gov.sg/infocentre/pdfs/tanker-vessel.pdf [Accessed 26 March 2006].

(27) Ng, W. H. and Baumgarten, S., 1998. *Guide to the petrochemical and chemical industry in Singapore.* 2nd ed. The Strategist Oil Report.

(28) Pitty, S. S., 2004. *Scheduling of serial multiproduct batch processes*. Thesis (M.Eng.). National University of Singapore.

(29) Rana, K. and Vickson, R.G., 1988. A model and solution algorithm for optimal routing of a time-chartered containership. *Transportation Science*, 22 (2), 83-95.

(30) Rana, K. and Vickson, R. G., 1991. Routing container ships using Lagrangean relaxation and decomposition. *Transportation Science*, 25 (3), 201-214.

(31) Sheppard, E. J. and Seidman, D., 2001. Ocean shipping alliances: the wave of the future? *International Journal of Maritime Economics*, 3, 351.

(32) Sundaramoorthy, A. and Karimi, I. A., 2005. A simpler better slot-based continuous-time formulation for short-term scheduling in multipurpose batch plants. *Chemical Engineering Science*, 60, 2679-2702.

(33) UNCTAD, 2004. *Review of Maritime Transport, 2004*. United Nations, New York and Geneva.

(34) Vis, I. F. A. and Koster, R., 2003. Transshipment of containers at a container terminal: an overview. *European Journal of Operational Research*, 147, 1-16.

(35) Warachka, M. and Karimi, I. A., 2004. A review of ship routing and scheduling optimization for maritime chemical transportation. Working paper. The logistics Institute-Asia Pacific, and the Department of Chemical & Biomolecular Engineering, National University of Singapore.

APPENDIX A: GAMS FILES FOR CHAPTER 4

A. 1 Formulation F1

```
Variables
x(j,k)          1 if cargo j IS transshipped during slot k
U(j)            1 if cargo j is serviced
Z(n,p,pc,k)     1 if carrier n is at position p in slot k and pc in slot k+1
y(j,j)          1 if cargo j is serviced later than cargo jc
DTS(n)          departure time for carrier n
DTB(m)          departure time for carrier m
TS(n,k)         start time of slot k in carrier n
T(j)            start time of serving cargo j
tx(j,k)         multiplication variable
delta(j,j)      dummy 0-1 variable
d(j,j,j)        dummy 0-1 variable
TTCC            total time cost of all ships;
Binary variable x,y;
Positive variable TS,Z, T, DTB, DTS, d, u, delta,tx;
Free variable TTCC;
x1(k,n)$kn(k,n)..                       sum(j$(js(j,n)),X(j,k))=e=1;
x2a(i,j)$(osc(i) and ij(i,j))..          sum(k$kj(k,j), x(j,k))=e=1;
x2b(i,j)$(tsc(i) and ij(i,j))..          sum(k$kj(k,j), x(j,k))=e=u(j);
x2c(i,j,jc)$(ij(i,j) and ij(i,jc) and tsc(i) and (ord(j) ne ord(jc))..  u(j)+u(jc)=e=1;
TS3(n,k)$kn(k,n)..                      TS(n,k)=g=sum(j$js(j,n), ETAC(j)*X(j,k));
z4a(n,p,k)$(kn(k,n) and ord(k)<kmaxn(n) and pn(p,n))..
sum(pc$pn(pc,n),Z(n,p,pc,k))=e=sum(j$(js(j,n) and jp(j,p)),X(j,k));
z4b(n,p,k)$(kn(k,n) and ord(k)<kmaxn(n) and pn(p,n))..
sum(pc$pn(pc,n),Z(n,pc,p,k))=e=sum(j$(js(j,n) and jp(j,p)),X(j,k+1));
TS5(n,k)$(kn(k,n) and ord(k)<kmaxn(n))..
TS(n,k+1)=g=TS(n,k)+sum(j$js(j,n),tao(j)*X(j,k))+sum((p,pc)$(PN(P,N) AND
PN(PC,N)), theta(n,p,pc)*Z(n,p,pc,k));
Tb6a(p,j,jc)$(ord(jc)> ord(j) and jp(j,p) and jp(jc,p) and not jj(j,jc))..
T(jc)=g=T(j)+tao(j)*u(j)-bigM*(1-y(j,jc));
Tb6b(p,j,jc)$(ord(jc)> ord(j) and jp(j,p) and jp(jc,p) and not jj(j,jc))..
T(j)=g=T(jc)+tao(jc)*u(jc)-bigM*y(j,jc);
Tb7a(j)..                        T(j)=g=etac(j)*u(j)+mt(j)*(1-u(j));
Tb7b(j)..                        T(j)=l=mt(j)+bigM*u(j);
L9a(j,n,k)$(kn(k,n) and js(j,n))..      T(j)=g=TS(n,k)-bigM*(1-X(j,k));
L9b(j,n,k)$(kn(k,n) and js(j,n))..      T(j)=l=TS(n,k)+bigM*(1-X(j,k));
L10(n)..
sum(j$js(j,n),t(j))=e=sum(k$kn(k,n),ts(n,k))+sum(j$js(j,n), (1-u(j))*mt(j));
L11a(n,k)$kn(k,n)..              TS(n,k)=e=sum(j$js(j,n),tx(j,k));
L11b(j)..                        T(j)=e=sum(k$kj(k,j),tx(j,k))+mt(j)*(1-u(j));
L11c(j,k)$kj(k,j)..              tx(j,k)=l=bigM*x(j,k);
dt12a(j,m)$jb(j,m)..             DTB(m)=g=T(j)+tao(j)*u(j);
dt12b(k,m,n)$kn(k,n) ..          DTB(m)=g=TS(n,k)+sum(j$(js(j,n) AND
JB(J,M)),tao(j)*X(j,k))-bigM*(1-sum(j$(jb(j,m) and js(j,n)),X(j,k)));
```

```
addy(i,ic,p,j,jc)$(ord(jc)>ord(j) and jp(j,p) and jp(jc,p) and ij(i,j) and ij(ic,jc) and not
jj(j,jc) and (tsc(i) or tsc(iC)))..   y(j,jc)+delta(j,jc)=e=u(jc)-u(j)+1;
addyy(p,j,jc,jcc)$(ord(jc) > ord(j) and ord(jcc)>ord(jc) and jp(j,p) and jp(jc,p) and
jp(jcc,p) and not jj(j,jc) and not jj(j,jcc) and not jj(jc,jcc))..
Y(j,jcc)+d(j,jc,jcc)=e=Y(j,jc)+Y(jc,jcc);
objfun.. TTCC=e=sum(m,TCCB(m)*(DTB(m)-ETAB(m)))+sum((n,k)$(kn(k,n) and
ord(k)=kmaxn(n)),TCCS(n)*(TS(n,k)+sum(j$js(j,n), tao(j)*X(j,k))-ETAS(n)));
u.fx(j)$(sum(i$ij(i,j),osc(i)))=1;
TS.lo(n,k)$kn(k,n)=MTS(n);
T.lo(j)$(sum(i$ij(i,j),Tsc(i)))=mt(j);
T.lo(j)$(sum(i$ij(i,j),Osc(i)))=ETAC(J);
DTS.lo(n)=mts(n);
DTB.lo(m)=smin(j$jb(j,m),mt(j));
Z.fx(n,p,pc,k)$((not pn(p,n)) or (not pn(pc,n)) and kn(k,n) and ord(k)<kmaxn(n))=0;
Z.up(n,p,pc,k)$(pn(p,n) AND pn(pc,n) and kn(k,n))=1;
u.up(j)=1;
d.up(j,j,j)=1;
delta.up(j,j)=1;
```

A. 2 Formulation F2 and F3

```
Variables
x(j,k)          1 if cargo j IS transshipped during slot k
U(j)            1 if cargo j is serviced
y(j,j)          1 if cargo j is serviced later than cargo jc
DTS(n)          departure time for carrier n
DTB(m)          departure time for carrier m
T(j)            start time of serving cargo j
TB(p,k)         start time of slot k in position p
TX(j,k)         multiplication variable
delta(j,j)      dummy 0-1 variable
d(j,j,j)        dummy 0-1 variable
TTCC            total time cost of all ships;
Binary variable x,y;
positive variable T,TB, DTB, DTS,u,d,delta,tx,tsx;
free variable TTCC;
x14(k,p)$kp(k,p)..                         sum(j$jp(j,p),X(j,k))=l=1;
x2a(i,j)$(osc(i) and ij(i,j))..            sum(k$kj(k,j), x(j,k))=e=1;
x2b(i,j)$(tsc(i) and ij(i,j))..            sum(k$kj(k,j), x(j,k))=e=u(j);
x2c(i,j,jc)$(ij(i,j) and ij(i,jc) and tsc(i) and (ord(j) ne ord(jc)))..  u(j)+u(jc)=e=1;
TB15a(p,k)$kp(k,p)..
TB(p,k)=g=sum(j$jp(j,p),etac(j)*x(j,k))+mtb(p)*(1-sum(j$jp(j,p),x(j,k)));
tb15b(p,k)$kp(k,p)..
TB(p,k)=l=mtb(p)+bigM*sum(j$jp(j,p),x(j,k));
tb16(p,k)$(kp(k,p) and ord(k)<kmaxp(p))..
TB(p,k+1)=g=TB(p,k)+sum(j$jp(j,p),tao(j)*X(j,k));
ts17a(n,j,jc)$(ord(jc)>ord(j) and js(j,n) and js(jc,n) and not jj(j,jc))..
T(jc)=g=T(j)+(tao(j)+sum((p,pc)$(jp(j,p) and jp(jc,pc)),theta(n,p,pc)))*u(j)-bigM*(1-
y(j,jc));
```

```
ts17b(n,j,jc)$(ord(jc)>ord(j) and js(j,n) and js(jc,n) and not jj(j,jc))..
T(j)=g=T(jc)+(tao(jc)+sum((p,pc)$(jp(j,p) and jp(jc,pc)),theta(n,pc,p)))*u(jc)-
bigM*y(j,jc);
ts7a(j)..                    T(j)=g=etac(j)*u(j)+(1-u(j))*mt(j);
ts7b(j)..                    T(j)=l=mt(j)+bigM*u(j);
L19a(p,j,k)$(kp(k,p) and jp(j,p))..T(j)=g=TB(p,k)-bigM*(1-X(j,k));
L19b(p,j,k)$(kp(k,p) and jp(j,p))..T(j)=l=TB(p,k)+bigM*(1-X(j,k));
L20(p).. sum(j$jp(j,p),t(j))+sum(k$kp(k,p),(1-
sum(j$jp(j,p),x(j,k)))*mtb(p))=e=sum(k$kp(k,p),tb(p,k))+sum(j$jp(j,p),(1-u(j))*mt(j));
L21a(p,k)$kp(k,p)..          tb(p,k)=e=sum(j$jp(j,p),tx(j,k))+(1-
sum(j$jp(j,p),x(j,k)))*mtb(p);
L11b(j)..                    T(j)=e=sum(k$kj(k,j),tx(j,k))+mt(j)*(1-u(j));
L11c(j,k)$kj(k,j)..          tx(j,k)=l=bigM*x(j,k);
DT22a(j,n)$js(j,n)..                        DTS(n)=g=T(j)+tao(j)*u(j);
DT22b(p,k,n)$kp(k,p)..                      DTS(n)=g=TB(p,k)+sum(j$(jp(j,p) and
js(j,n)),tao(j)*X(j,k))-bigM*(1-sum(j$(js(j,n) and jp(j,p) ),X(j,k)));
DT23(p,m,k)$(kp(k,p) and ord(k)=kmaxp(p) and pm(p,m))..
DTB(m)=g=TB(p,k)+sum(j$jp(j,p), tao(j)*X(j,k));
addx(p,k)$(kp(k,p) and ord(k)<kmaxp(p))..
sum(j$jp(j,p),X(j,k+1))=g=sum(j$jp(j,p),X(j,k));
addy(i,ic,n,j,jc)$(ord(jc)>ord(j) and js(j,n) and js(jc,n) and ij(i,j) and ij(ic,jc) and not
jj(j,jc) and (tsc(i) or tsc(iC)))..   y(j,jc)+delta(j,jc)=e=u(jc)-u(j)+1;
addyy(n,j,jc,jcc)$(ord(jc) > ord(j) and ord(jcc)>ord(jc) and js(j,n) and js(jc,n) and
js(jcc,n) and not jj(j,jc) and not jj(j,jcc) and not jj(jc,jcc))..
Y(j,jcc)+d(j,jc,jcc)=e=Y(j,jc)+Y(jc,jcc);
objfun..  TTCC=e=sum(m,TCCB(m)*(DTB(m)-
ETAB(m)))+sum(n,TCCS(n)*(DTS(n)-ETAS(n)));
u.fx(j)$(sum(i$ij(i,j),osc(i)))=1;
T.lo(j)$(sum(i$ij(i,j),Tsc(i)))=mt(j);
T.lo(j)$(sum(i$ij(i,j),Osc(i)))=ETAC(J);
TB.lo(p,k)$kp(k,p)=mtb(p);
DTS.lo(n)=mts(n);
DTB.lo(m)=smin(j$jb(j,m),mt(j));
u.up(j)=1;
d.up(j,j,j)=1;
delta.up(j,j)=1;
```

APPENDIX B: GAMS FILE FOR CHAPTER 6

```
Variables
x(n,k,j)        1 if lightering vessel n serves cargo j during slot k
y(j,jc)         1 if cargo jc is served sometime after cargo j in position p
ze(n,k)         1 if slot k is the last slot of a voyage
Z(n,k,j,jc)     1 if j served in slot k and jc in slot k+1 by vessel n in the same voyage
U(n,k)          1 if slot k of ship n is used
VP(n,k,j)       amount of cargo j that lightering vessel n lighters during slot k
VT(n,k)         accumulated volume of crude oil on vessel n during slot k
TS(n,k)         time at which vessel n starts slot k
T(j)            time when cargo j starts transfer
DTB(m)          time at which m departs
TD(n)           total distance traveled by vessel n
ATR(n,k,j)      arrival time at refinery of cargo j served during slot k by vessel n
ATRJ(j)         arrival time at refinery of cargo j served during slot k by vessel n
NC(n,k)         number of comparments used during slot k of vessel n
NT(n,k)         accumulated number of compartments used in slot k of vessel n
VT(n,k)         accumulated volume at the end of slot k of small ship n
NT(n,k)         accumulated number of compartments at the end of slot k of small ship
n
WT(n,k)
VZ(n,k)         accumulated volume at the end of slot k of small ship n
NZ(n,k)         accumulated number of compartments at the end of slot k of small ship
n
WZ(n,k)
delta(j,jc)     0-1 continuous dummy variable
d(j,jc,jcc)     0-1 continuous dummy variable
TC              total cost of the system;
Binary variable  x,y,ze;
Integer variable nc;
Positive variable z,vp,ts,t,dtb, td, tr, delta, d,u, WT, NT, VT, WZ, NZ, VZ,atr, atrj;
Free variable    TC;
u1(n,k)$kn(k,n)..                      u(n,k)=e=sum(j$jn(j,n),x(n,k,j));
u2(n,k)$(kn(k,n) and ord(k)<kmax(n)).. u(n,k+1)=l=u(n,k);
x3a(m)..                               sum((n,k,j)$(jn(j,n) and jm(j,m) and
kn(k,n)),x(n,k,j))=g=nsl(m);
x3b(m)..                               sum((n,k,j)$(jn(j,n) and jm(j,m) and
kn(k,n)),x(n,k,j))=l=nsu(m);
z4(n,k)$kn(k,n)..                      ze(n,k)=l=u(n,k);
z5a(j,n,k)$(jn(j,n) and kn(k,n)and ord(k)<kmax(n))..
        x(n,k,j)=g=sum(jc$(jn(jc,n) and jj(j,jc)),z(n,k,j,jc));
z5b(jc,n,k)$(jn(jc,n) and kn(k,n)and ord(k)<kmax(n))..
        x(n,k+1,jc)=g=sum(j$(jn(j,n) and jj(j,jc)),z(n,k,j,jc));
z5c(n,k)$(kn(k,n)and ord(k)<kmax(n))..
        1-ze(n,k)=g=sum((j,jc)$(jn(j,n) and jn(jc,n) and jj(j,jc)),z(n,k,j,jc));
z5d(j,jc,n,k)$(jn(j,n) and jn(jc,n) and jj(j,jc)  and kn(k,n)and ord(k)<kmax(n))..
        z(n,k,j,jc)=g=x(n,k,j)+x(n,k+1,jc)-ze(n,k)-1;
```

```
z6(n)..  sum(k$kn(k,n),ze(n,k))=e=sum(k$kn(k,n),u(n,k))-sum((k,j,jc)$(kn(k,n) and
ord(k)<kmax(n) and jn(j,n) and jn(jc,n) and jj(j,jc)),z(n,k,j,jc));
x7(j,m)$(jm(j,m) and nsu(m)>2)..         sum((n,k)$(kn(k,n) and jn(j,n)),x(n,k,j))=l=1;
v8a(n,j,k)$(kn(k,n) and jn(j,n))..       vp(n,k,j)=l=x(n,k,j)*min(nu(n)*sc(n),
wd(j,n)/rho(j), sum(c$cj(c,j),ps(c)));
v8b(c)$imc(c)..                          sum((n,j,k)$(kn(k,n) and jn(j,n) and
cj(c,j)),vp(n,k,j))=l=ps(c);
n9a(n,k)$kn(k,n)..       sum(j$jn(j,n),vp(n,k,j))=l=nc(n,k)*sc(n);
n9b(n,k)$kn(k,n)..       sum(j$jn(j,n),vp(n,k,j))=g=(nc(n,k)-1)*sc(n);
v10(m)..                 sum((n,j,k)$(jm(j,m) and jn(j,n) and
kn(k,n)),vp(n,k,j)*rho(j))=e=lw(m);
w11b(n,k)$(ord(k)=1)..   wt(n,k)=e=sum(j$jn(j,n), rho(j)*vp(n,k,j));
v12b(n,k)$(ord(k)=1)..   vt(n,k)=e=sum(j$jn(j,n), vp(n,k,j));
n13b(n,k)$(ord(k)=1)..   nt(n,k)=e=nc(n,k);
w14a(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wt(n,k+1)=g=wt(n,k)+sum(j$jn(j,n),rho(j)*vp(n,k+1,j))-wtu(n)*ze(n,k);
w14b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wt(n,k+1)=l=wt(n,k)+sum(j$jn(j,n),rho(j)*vp(n,k+1,j));
w14c(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wt(n,k+1)=l=sum(j$jn(j,n),rho(j)*vp(n,k+1,j))+wtu(n)*(1-ze(n,k));
w14d(n,k)$(kn(k,n) and ord(k)>1)..       wt(n,k)=g=sum(j$jn(j,n),rho(j)*vp(n,k,j));
v16a(n,k)$(kn(k,n) and ord(k)<kmax(n))..
vt(n,k+1)=g=vt(n,k)+sum(j$(jn(j,n)),vp(n,k+1,j))-vtu(n)*ze(n,k);
v16b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
vt(n,k+1)=l=sum(j$jn(j,n),vp(n,k+1,j))+vtu(n)*(1-ze(n,k));
v16c(n,k)$(kn(k,n) and ord(k)<kmax(n))..
vt(n,k+1)=l=vt(n,k)+sum(j$jn(j,n),vp(n,k+1,j));
v16d(n,k)$(kn(k,n) and ord(k)>1)..       vt(n,k)=g=sum(j$(jn(j,n)),vp(n,k,j));
n18a(n,k)$(kn(k,n) and ord(k)<kmax(n)).. nt(n,k+1)=g=nt(n,k)+nc(n,k+1)-
nu(n)*ze(n,k);
n18b(n,k)$(kn(k,n) and ord(k)<kmax(n)).. nt(n,k+1)=l=nt(n,k)+nc(n,k+1);
n18c(n,k)$(kn(k,n) and ord(k)<kmax(n)).. nt(n,k+1)=l=nc(n,k+1)+nu(n)*(1-ze(n,k));
n18d(n,k)$(kn(k,n) and ord(k)>1)..       nt(n,k)=g=nc(n,k);
w20(n,k)$kn(k,n)..       wt(n,k)=l=sum(j$jn(j,n),wd(j,n)*x(n,k,j));
n21(n,k)$kn(k,n)..       nt(n,k)=l=nu(n)*u(n,k);
t22a(n,k)$(kn(k,n) and ord(k)<kmax(n))..
ts(n,k+1)=g=ts(n,k)+sum(j$(jn(j,n)),(mdt(j)*x(n,k,j))+vp(n,k,j)/fin(n))+sum((j,jc)$(jn(
j,n) and jn(jc,n) and jj(j,jc)),tpp(n,j,jc)*z(n,k,j,jc));
t22b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
ts(n,k+1)=g=ts(n,k)+sum(j$jn(j,n),(tpr(n,j)+tra(n,j)+mdt(j)+dtr(j))*x(n,k,j)+vp(n,k,j)/f
in(n))+vt(n,k)/fout(n)+sum(j$jn(j,n), tap(n,j)*x(n,k+1,j))-bigMT*(1-ze(n,k));
t23(n,k)$kn(k,n)..                       ts(n,k)=g=sum(j$(jn(j,n)),eta(j)*x(n,k,j));
t24(n,k)$(kn(k,n) and ord(k)=1)..
ts(n,k)=g=etas(n)+sum(j$(jn(j,n)),tap(n,j)*x(n,k,j));
t25(mc,n,k)$kn(k,n)..                    ts(n,k)=g=sum(m$mm(m,mc), dtb(m)+toa(m))-
bigMT*(1-sum(j$(jn(j,n) and jm(j,mc) and s2(j)),x(n,k,j)));
a26a(m,n,j,k)$(jn(j,n) and kn(k,n)and jm(j,m) and nsu(m)<3)..
        atr(n,k,j)=g=ts(n,k)+(mdt(j)+tpr(n,j))*x(n,k,j)+vp(n,k,j)/fin(n)-bigMT*(2-
ze(n,k)-x(n,k,j));
a26b(m,n,j,k)$(jn(j,n) and kn(k,n)and jm(j,m) and nsu(m)>2)..
```

```
        atrj(j)=g=ts(n,k)+(mdt(j)+tpr(n,j))*x(n,k,j)+vp(n,k,j)/fin(n)-bigMT*(2-ze(n,k)-
x(n,k,j));
a27a(m,mc,n,j,jc,k)$(kn(k,n) and ord(k)<kmax(n) and jn(j,n) and jn(jc,n) and jj(j,jc)
and jm(j,m) and jm(jc,mc) and nsu(m)<3 and nsu(mc)<3)..
        atr(n,k,j)=g=atr(n,k+1,jc)-bigMT*(1-z(n,k,j,jc));
a27b(m,mc,n,j,jc,k)$(kn(k,n) and ord(k)<kmax(n) and jn(j,n) and jn(jc,n) and jj(j,jc)
and jm(j,m) and jm(jc,mc) and nsu(m)<3 and nsu(mc)>2)..
        atr(n,k,j)=g=atrj(jc)-bigMT*(1-z(n,k,j,jc));
a27c(m,mc,n,j,jc,k)$(kn(k,n) and ord(k)<kmax(n) and jn(j,n) and jn(jc,n) and jj(j,jc)
and jm(j,m) and jm(jc,mc) and nsu(m)>2 and nsu(mc)<3)..
        atrj(j)=g=atr(n,k+1,jc)-bigMT*(1-z(n,k,j,jc));
a27d(m,mc,j,jc)$(jj(j,jc) and jm(j,m) and jm(jc,mc) and nsu(m)>2 and nsu(mc)>2)..
        atrj(j)=g=atrj(jc)-bigMT*(1-sum((n,k)$(kn(k,n) and ord(k)<kmax(n) and jn(j,n)
and jn(jc,n)), z(n,k,j,jc)));
t28a(m,p,j,jc)$(ord(j)<ord(jc) and jp(j,p) and jp(jc,p) and jm(j,m) and nsu(m)>2 and
not sum(c, (cj(c,j) and cj(c,jc))))..
        t(jc)=g=t(j)+sum((n,k)$(kn(k,n) and jn(j,n)), vp(n,k,j)/fin(n)+mdt(j)*x(n,k,j))-
bigMT*(1-y(j,jc));
t28b(m,p,j,jc)$(ord(j)<ord(jc) and jp(j,p) and jp(jc,p) and jm(j,m) and nsu(m)>2 and
not sum(c, (cj(c,j) and cj(c,jc))))..
        t(j)=g=t(jc)+sum((n,k)$(kn(k,n) and jn(jc,n)),
vp(n,k,jc)/fin(n)+mdt(jc)*x(n,k,jc))-bigMT*y(j,jc);
t28c(m,p,j,jc)$(ord(j)<ord(jc) and jp(j,p) and jp(jc,p) and jm(j,m) and nsu(m)>2 and
sum(c, (cj(c,j) and cj(c,jc))))..
        t(jc)=g=t(j)+sum((n,k)$(kn(k,n) and jn(j,n)), vp(n,k,j)/fin(n)+mdt(j)*x(n,k,j));
t29a(m,j)$(jm(j,m) and nsu(m)>2)..
        t(j)=g=sum((n,k)$(kn(k,n) and jn(j,n)), (etas(n)+tap(n,j))*x(n,k,j));
t29b(m,j)$(jm(j,m) and nsu(m)>2)..
        t(j)=l=eta(j)+bigMT*sum((k,n)$(kn(k,n) and jn(j,n)), x(n,k,j));
tt30a(m,j,n,k)$(kn(k,n) and jn(j,n) and jm(j,m) and nsu(m)>2)..
        t(j)=g=ts(n,k)-bigMT*(1-x(n,k,j));
tt30b(m,j,n,k)$(kn(k,n) and jn(j,n) and jm(j,m) and nsu(m)>2)..
        t(j)=l=ts(n,k)+bigMT*(1-x(n,k,j));
add31(m,p,j,jc)$(ord(j)<ord(jc) and jp(j,p) and jp(jc,p) and jm(j,m) and nsu(m)>2)..
        y(j,jc)+delta(j,jc)=e=sum((k,n)$(kn(k,n) and jn(jc,n)), x(n,k,jc))-
sum((k,n)$(kn(k,n) and jn(j,n)), x(n,k,j))+1;
add32(m,p,j,jc,jcc)$(ord(j)<ord(jc) and ord(jc)<ord(jcc) and jp(j,p) and jp(jc,p)and
jp(jcc,p) and jm(j,m) and nsu(m)>2)..
        y(j,jcc)+d(j,jc,jcc)=e=y(j,jc)+y(jc,jcc);
d33a(m,n,k)$kn(k,n)..
        dtb(m)=g=ts(n,k)+sum(j$(jn(j,n) and jm(j,m)),mdt(j)*x(n,k,j)+vp(n,k,j)/fin(n))-
bigMT*(1-sum(j$(jn(j,n) and jm(j,m)),x(n,k,j)));
d33b(m,j)$(jm(j,m) and nsu(m)>2)..
        dtb(m)=g=t(j)+sum((n,k)$(kn(k,n) and jn(j,n)), mdt(j)*x(n,k,j)+vp(n,k,j)/fin(n));
d34(n).. td(n)=e=sum((j,k)$(jn(j,n) and kn(k,n)),2*dr(j)*x(n,k,j))-sum((j,jc,k)$(jn(j,n)
and jn(jc,n) and jj(j,jc) and kn(k,n) and ord(k)<kmax(n)),z(n,k,j,jc)*(2*dr(j)-
tpp(n,j,jc)*vf(n)));
wz1a(n,k)$(kn(k,n) and ord(k)=1)..
sum(j$jn(j,n),rho(j)*vp(n,k,j))=l=sum(j$jn(j,n),wd(j,n)*x(n,k,j));
```

```
wz1b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wz(n,k)+sum(j$jn(j,n),rho(j)*vp(n,k+1,j))=l=sum(j$jn(j,n),wd(j,n)*x(n,k+1,j));
wz2a(n,k)$(kn(k,n) and ord(k)=1)..       wz(n,k)=l=sum(j$jn(j,n),rho(j)*vp(n,k,j));
wz2b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wz(n,k+1)=l=wz(n,k)+sum(j$jn(j,n),rho(j)*vp(n,k+1,j));
wz3(n,k)$(kn(k,n))..                     wz(n,k)=l=wtu(n)*(1-ze(n,k));
wz4a(n,k)$(kn(k,n) and ord(k)=1)..
wz(n,k)+wtu(n)*ze(n,k)=g=sum(j$jn(j,n),rho(j)*vp(n,k,j));
wz4b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
wz(n,k+1)+wtu(n)*ze(n,k+1)=g=wz(n,k)+sum(j$jn(j,n),rho(j)*vp(n,k+1,j));
nz1a(n,k)$(kn(k,n) and ord(k)=1)..       nc(n,k)=l=nu(n)*u(n,k);
nz1b(n,k)$(kn(k,n) and ord(k)<kmax(n)).. nz(n,k)+nc(n,k+1)=l=nu(n)*u(n,k+1);
nz2a(n,k)$(kn(k,n) and ord(k)=1)..       nz(n,k)=l=nc(n,k);
nz2b(n,k)$(kn(k,n) and ord(k)<kmax(n)).. nz(n,k+1)=l=nz(n,k)+nc(n,k+1);
nz3(n,k)$(kn(k,n))..                     nz(n,k)=l=nu(n)*(1-ze(n,k));
nz4a(n,k)$(kn(k,n) and ord(k)=1)..       nz(n,k)+nu(n)*ze(n,k)=g=nc(n,k);
nz4b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
nz(n,k+1)+nu(n)*ze(n,k+1)=g=nz(n,k)+nc(n,k+1);
vz1a(n,k)$(kn(k,n) and ord(k)<kmax(n) and ord(k)=1)..
ts(n,k+1)=g=ts(n,k)+sum(j$jn(j,n),(tpr(n,j)+tra(n,j)+mdt(j)+dtr(j))*x(n,k,j)+vp(n,k,j)/f
in(n))+sum(j$jn(j,n), vp(n,k,j))/fout(n)+sum(j$jn(j,n), tap(n,j)*x(n,k+1,j))-bigMT*(1-
ze(n,k));
vz1b(n,k)$(kn(k,n) and ord(k)<kmax(n)-1)..
ts(n,k+2)=g=ts(n,k+1)+sum(j$jn(j,n),(tpr(n,j)+tra(n,j)+mdt(j)+dtr(j))*x(n,k+1,j)+vp(n,
k+1,j)/fin(n))+(vz(n,k)+sum(j$jn(j,n),vp(n,k+1,j)))/fout(n)+sum(j$jn(j,n),
tap(n,j)*x(n,k+2,j))-bigMT*(1-ze(n,k+1));
vz2a(n,k)$(kn(k,n) and ord(k)=1)..                vz(n,k)=l=sum(j$jn(j,n), vp(n,k,j));
vz2b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
vz(n,k+1)=l=(vz(n,k)+sum(j$jn(j,n),vp(n,k+1,j)));
vz3(n,k)$(kn(k,n))..                              vz(n,k)=l=vtu(n)*(1-ze(n,k));
vz4a(n,k)$(kn(k,n) and ord(k)=1)..
vz(n,k)+vtu(n)*ze(n,k)=g=sum(j$jn(j,n), vp(n,k,j));
vz4b(n,k)$(kn(k,n) and ord(k)<kmax(n))..
vz(n,k+1)+vtu(n)*ze(n,k+1)=g=(vz(n,k)+sum(j$jn(j,n),vp(n,k+1,j)));
objfun.. TC=e=sum(m$(not tst(m)),dc(m)*(dtb(m)-etab(m)-
ad(m)))+sum((m,mc)$mm(m,mc), dc(m)*(dtb(mc)-etab(m)-toa(m)-
ad(m)))+sum(n,td(n)*fc(n))+sum((k,n)$kn(k,n),vc(n)*ze(n,k))+sum((m,n,j,k)$(kn(k,n)
and jn(j,n) and jm(j,m) and nsu(m)<3),ddp(j)*(atr(n,k,j)-dd(j)))+sum((m,j)$(jm(j,m)
and nsu(m)>2),ddp(j)*(atrj(j)-dd(j)));
ts.lo(n,k)$kn(k,n)=etas(n);
t.lo(j)$sum(m$(jm(j,m) and nsu(m)>2),jm(j,m))=eta(j);
dtb.lo(m)=etab(m)+toa(m)+ad(m);
atr.lo(n,k,j)$(kn(k,n) and jn(j,n))=DD(j);
atrj.lo(j)=DD(j);
z.up(n,k,j,jc)$(jn(j,n) and jn(jc,n) and jj(j,jc)  and kn(k,n)and ord(k)<kmax(n))=1;
u.up(n,k)=1;
delta.up(j,jc)=1;
d.up(j,jc,jcc)=1;
y.fx(j,jc)$(sum(p, (jp(j,p) and jp(jc,p))) and sum(c, (cj(c,j) and cj(c,jc))) and
ord(j)<ord(jc) and sum(m, (jm(j,m) and jm(jc,m) and nsu(m)>2)))=1;
```

APPENDIX C: REDUCED FORMULATIONS FOR CHAPTER 6

C.1 Formulation O1

$$U_{nk} = \sum_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (6.1)$$

$$NS_m^L \le \sum_n \sum_k \sum_j x_{nkj} \le NS_m^U \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.3)$$

$$VP_{nkj} \le x_{nkj} \min[\frac{WD_{nj}}{\rho_j}, PS_c] \qquad j \in \boldsymbol{JS}_n \qquad (6.8a)$$

$$\sum_n \sum_k \sum_j VP_{nkj} \le PS_c \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JC}_c, c \ni PS_c \le LW_m/\rho_c \qquad (6.8b)$$

$$\sum_n \sum_k \sum_j VP_{nkj}\rho_j = LW_m \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.10)$$

$$\begin{aligned} TS_{n(k+1)} \ge TS_{nk} &+ \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + \sum_j TPR_{nj} x_{nkj} \\ &+ \left(\sum_j DTR_j x_{nkj} + \frac{VP_{nk}}{FOUT_n} \right) + \sum_j \left(TRA_{nj} x_{nkj} \right) \end{aligned} \qquad j \in \boldsymbol{JS}_n, k < K_n \qquad (6.22b)$$

$$TS_{nk} \ge \sum_j ETA_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (6.23)$$

$$DTB_m \ge TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) - HT \left(1 - \sum_j x_{nkj} \right) \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.33a)$$

$$TC = \sum_n \sum_{k=1}^{K_n} VC_n U_{nk} + \sum_m TCC_m \left(DTB_m - ETAB_m \right) \qquad (6.35a)$$

C.2 Formulation O2

$$U_{nk} = \sum_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (6.1)$$

$$NS_m^L \le \sum_n \sum_k \sum_j x_{nkj} \le NS_m^U \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.3)$$

$$VP_{nkj} \le x_{nkj} \min[N_n^U SC_n, \frac{WD_{nj}}{\rho_j}, PS_c] \qquad j \in \boldsymbol{JS}_n \qquad (6.8a)$$

$$\sum_n \sum_k \sum_j VP_{nkj} \le PS_c \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JC}_c, c \ni PS_c \le LW_m/\rho_c \qquad (6.8b)$$

$$\sum_n \sum_k \sum_j VP_{nkj}\rho_j = LW_m \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.10)$$

$$TS_{n(k+1)} \geq TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) + \sum_j TPR_{nj} x_{nkj}$$

$$+ \left(\sum_j DTR_j x_{nkj} + \frac{VP_{nk}}{FOUT_n} \right) + \sum_j \left(TRA_{nj} x_{nkj} \right) \qquad j \in \boldsymbol{JS}_n,\ k < K_n \qquad (6.22b)$$

$$TS_{nk} \geq \sum_j ETA_j x_{nkj} \qquad j \in \boldsymbol{JS}_n \qquad (6.23)$$

$$TS_{n1} \geq ETAS_n + \sum_j TAP_{nj} x_{n1j} \qquad j \in \boldsymbol{JS}_n \qquad (6.24)$$

$$TS_{nk} \geq DTB_m + TOA_m - HT \left(1 - \sum_j x_{nkj} \right)$$

$$(m, m'(j)) \in \boldsymbol{MM},\ j \in \boldsymbol{JS}_n \cap \boldsymbol{S2} \cap \boldsymbol{JB}_{m'} \qquad (6.25)$$

$$DTB_m \geq TS_{nk} + \sum_j \left(MDT_j x_{nkj} + \frac{VP_{nkj}}{FIN_{nj}} \right) - HT \left(1 - \sum_j x_{nkj} \right) \qquad j \in \boldsymbol{JS}_n \cap \boldsymbol{JB}_m \qquad (6.33a)$$

$$TC = \sum_n \sum_{k=1}^{K_n} VC_n U_{nk} + \sum_m TCC_m \left(DTB_m - ETAB_m \right) \qquad (6.35a)$$