University of South Florida

DIGITAL COMMONS
@ UNIVERSITY OF SOUTH FLORIDA

Digital Commons @ University of South Florida

USF Tampa Graduate Theses and Dissertations

USF Graduate Theses and Dissertations

7-1-2004

Robust Dock Assignments at Less-Than-Truckload Terminals

Mesut Korhan Acar
University of South Florida

Follow this and additional works at: https://digitalcommons.usf.edu/etd

Part of the American Studies Commons

Scholar Commons Citation
Acar, Mesut Korhan, "Robust Dock Assignments at Less-Than-Truckload Terminals" (2004). *USF Tampa Graduate Theses and Dissertations.*
https://digitalcommons.usf.edu/etd/935

This Thesis is brought to you for free and open access by the USF Graduate Theses and Dissertations at Digital Commons @ University of South Florida. It has been accepted for inclusion in USF Tampa Graduate Theses and Dissertations by an authorized administrator of Digital Commons @ University of South Florida. For more information, please contact digitalcommons@usf.edu.

Robust Dock Assignments at Less-Than-Truckload Terminals

by

Mesut Korhan Acar

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Industrial Engineering
Department of Industrial and Management Systems Engineering
College of Engineering
University of South Florida

Major Professor: Ali Yalcin, Ph.D.
Jose L. Zayas-Castro, Ph.D.
Suresh Khator, Ph.D.

Date of Approval:
July 1, 2004

Keywords: Mixed integer quadratic model, gate assignment, assignment heuristic, real time assignment, efficiency improvement in transportation, terminal operations

© Copyright 2004, Mesut Korhan Acar

DEDICATION

To my family

ACKNOWLEDGEMENTS

I would like to thank my Dr. Ali Yalcin for his support and mentoring throughout the course of this research.

I would like to thank my committee members Dr. Jose Zayas-Castro and Dr. Suresh Khator for their kind reviews and support, and for the wisdom they possess which they allowed me to benefit.

Last but not the least, I would like to thank my family and friends, without them the journey would neither start nor complete.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ROBUST DOCK ASSIGNMENTS AT LESS-THAN-TRUCKLOAD TERMINALS

Mesut Korhan Acar

ABSTRACT

Less-than-truckload industry has a valuable potential for applications of operations research in two areas, network design and efficiency improvement within existing networks. This thesis focuses on the latter, specifically the less-than-truckload terminals where cross docking operations occur.

The assignment of incoming trailers to inbound docks is one of the critical decisions that affect the performance of less-than-truckload terminals. This research reviews existing models in literature and introduces an optimal mixed integer quadratic model with the objective of generating assignments that are robust against variability in system parameters such as truck arrival and service times, terminal characteristics and trailer load content. The computational limitations of the optimal model are discussed.

A dock assignment heuristic is developed to overcome the computational difficulties reported with the optimal model to solve realistic size problems. It is concluded that the heuristic is generally applicable and is robust against system variably. A dynamic dock assignment heuristic is later introduced to implement the decision process at real time. It is concluded that he dynamic dock assignment heuristic is also robust against system variability.

The last part presents a case study that benchmarks the dynamic dock assignment heuristic and existing static assignments at a real terminal. The results show that the dynamic dock assignment heuristic outperforms the static assignment under system variability. Conclusions and future research areas are finally addressed in the last chapter.

CHAPTER 1
INTRODUCTION

1.1 The Transportation Sector

The transportation sector, constituting 10% of service sector, is reported to contribute 80% of US GDP in 2002, utilizes 3.9 million miles of public road, 190 thousand miles of railroad track, 25 thousand miles of waterways, 145 major ports and 5000 airports to serve 4 trillion passenger miles and 3.7 trillion ton-miles of domestic freight annually [1]. According to US Census Bureau [2], truck transportation and warehousing industries generated a revenue of 237,485 million dollars in 2002, of which 13% was less-than-truckload (LTL) transportation and 6.5% was warehousing and storage. These numbers express the importance of less than truckload freight transportation segment.

1.2 Less-Than-Truckload (LTL) Segment

By definition, truckload (TL) motor carriers are those operating with loads, whose weight is either in excess of 10,000 lbs or whose load allows no other load to be carried, whereas less-than-truckload (LTL) carriers carry shipments with an actual weight of 10,000 lbs or less per destination [3, 4]. The efficiency of truckload shipment is closely related with its utilization level which starts to decline as the load starts to comprise a small fraction of a truck.

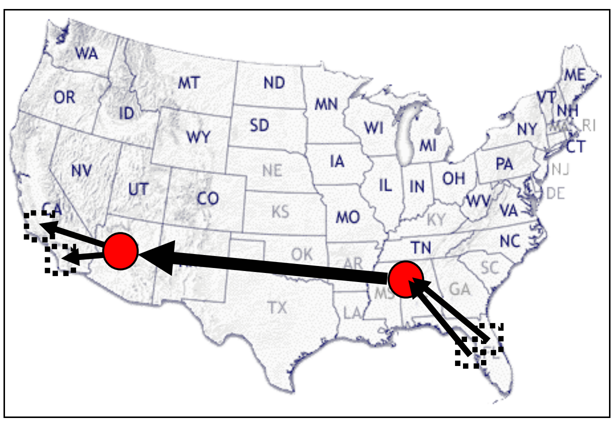

Figure 1 Example of LTL Shipment Consolidation

LTL carriers alleviate efficiency loss problem by consolidating shipments. LTL trailers can carry an average of 20-30 shipments that may have different origins and destinations. For example, a half truck load of goods originating from Tampa, FL is heading to San Diego, CA and another half truck load originating from Orlando, FL is bound to Los Angeles, CA. Instead of driving the trucks from coast to coast half empty, it would be a good idea to drive shipments a little out of the way to Birmingham, AL and consolidate the shipments in one truck which would then drive to west as depicted in Figure 1. This would incur some extra driving but would reduce the number of required trucks, trailers, drivers and total driving time. Birmingham, AL would also be an appropriate hub location for west bound freight coming from other locations in south east.

LTL industry makes the above scenario feasible by utilizing hub-and-spoke networks. These hubs are in the form of breakbulk terminals which are used to consolidate LTL shipments moving between end-of-line terminals at origin and destination points. The primary function of hub-and-spoke networks and breakbulk activity is to aggregate loads with common destinations to achieve economies of scale [5, 6].

1.2.1 LTL Networks

The flow of freight in hub-and-spoke networks is depicted in Figure 2. The shipment is picked up from the shipper and brought to the origin terminal, the entry point of the hub-and-spoke system. From the terminal, the freight is sent to a regional hub, where it is sorted and combined with other shipments, and sent on to other hubs where it is sorted again. Finally the order is sent to the destination terminal, which is the exit point of the hub-and-spoke system. From here, a local truck takes the shipment to its final destination. Handling freight at these terminals is labor intensive. Some carriers hold down the labor costs by restricting the sizes and weights of the packages which enables some automation via the use of draglines and conveyors.

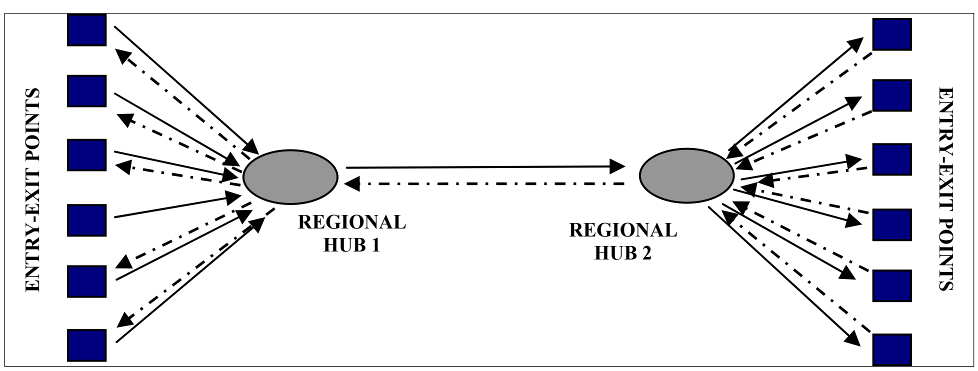

Figure 2 Hub-And-Spoke Network

There are mainly two important aspects of designing an LTL network. First is determining the number and locations of terminals and hubs. Appropriately located terminals and hubs in sufficient quantities can reduce costs significantly; however this task is difficult. A small number of terminals mean higher circuitry but fewer freight transfers and a large number means less circuitry but more freight transfers. The problem gets even more complicated from a global standpoint relative to freight demand and transshipment density, however, this aspect is out of the scope of this research. Some work in literature in this area are, an integer programming approach to hub location problem [7] and methods of reducing the number of constraints and variables [8-10].

The second aspect of an LTL network design is operational; i.e., efficiently moving freight between and inside terminals. This research relates to operational matters.

1.2.2 LTL Terminals

In a typical LTL network, terminals serve three basic functions, namely the out-bound, inbound, and breakbulk operations. These functions usually comprise unloading, sorting, consolidating and loading operations as depicted in Figure 3. The hubs, in addition to inbound and outbound operations, perform breakbulk operations which consolidate freight from end-of-line terminals and transfer the freight to other hubs. The time and influx of the circulation of freight in and out of terminals and hubs vary through the day.

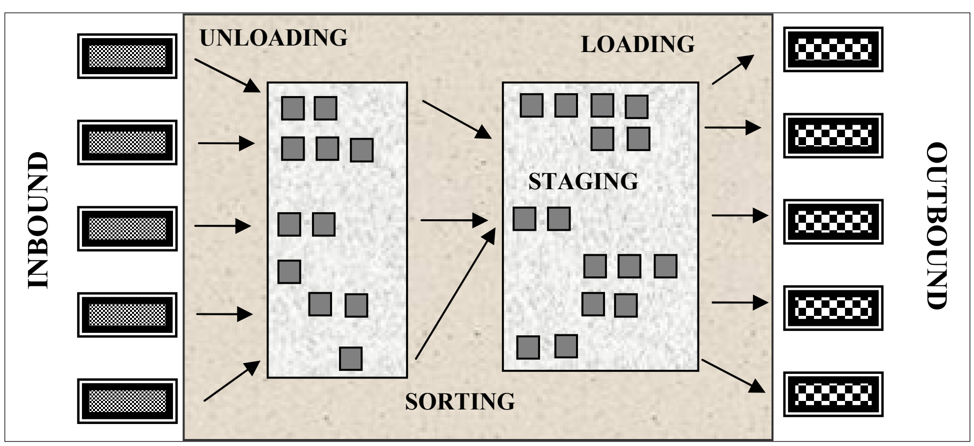

Figure 3 Operations in an LTL Terminal

Breakbulk operations are labor intensive and costly operations. A large LTL carrier can spend $300-$500 million annually handling freight (about 20% of total costs) [11]. Freight handling is important because time that a shipment spends at the terminal is wasted in the sense that the shipment is not making progress toward its destination. In some cases, rapid turnaround in the terminal can mean the difference between providing overnight or second-day service to a destination.

Depending on the size of the terminal, a typical breakbulk terminal workforce consists of an operations manager, supervisors and workers. When a trailer arrives at a terminal, it is either assigned to an inbound door or it is sent to a queue of trailers waiting to be unloaded. Once the trailer is parked at an open door, the shipments are unloaded and delivered to outbound doors according to their destinations. The unloaded trailer is removed from the inbound door and replaced by another incoming trailer. At the outbound doors, once a trailer is loaded, it is closed and replaced with another empty trailer that will be loaded with shipments to the same destination.

The assignment of doors followed by unloading, sorting and loading is called *cross-docking* operations. Material flows from inbound docks to outbound docks. Cross-docking is an operating strategy that moves items through without putting them into storage. Some recent work in cross docking systems is done by Gue and Bartholdi [11] and Gue [12]. Bartholdi and Gue [13] studied the optimal layout of a cross dock terminal. Cross-docking centers differ from conventional distribution centers because significant inventories are not accumulated for long times. Material is unloaded from trucks and immediately reloaded onto another vehicle by means of individual transportation units such as forklifts. Cross-dock operations thus play a critical role in reducing costs and delivery times through prevention of accumulation of inventory.

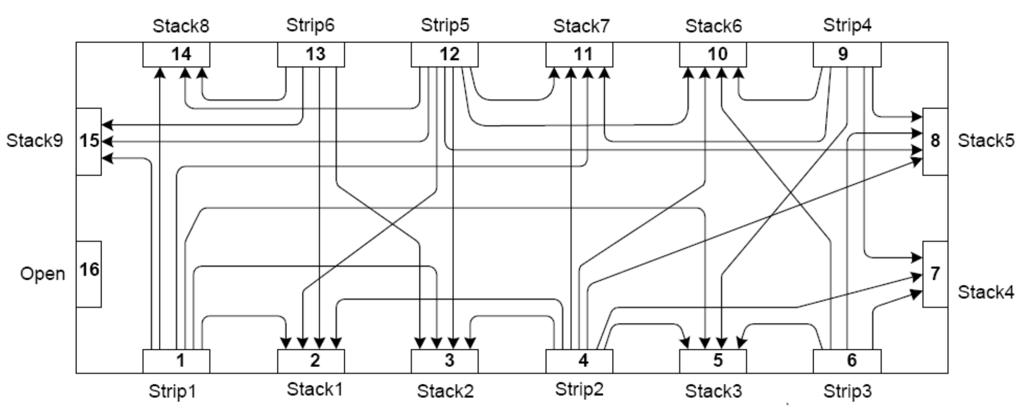

Figure 4 Crossdocking Terminal

LTL crossdock terminals, as depicted in Figure 4, resemble a warehouse with docks along its perimeter with 10 to 200 or more doors. The arrows show crossdocking routes. The two types of doors in the terminals are strip doors for receiving (unloading) and stack doors for shipping (loading) to specific destinations. Some doors can be assigned either type and are left open for that purpose, like door number 16 in Figure 4.

1.2.2.1 LTL Terminal Door (Dock) Assignment

The assignment of incoming and outgoing trailers to a dock (a terminal strip/stack door will be referred to as a dock in the remaining part of this thesis) is one of the critical decision factors that affect the performance of LTL terminals. The dock assignment problem is a resource allocation problem that is commonly addressed in the literature as a quadratic assignment problem. This problem is some ways similar to the gate assignment problem at airports. There are unique characteristics associated with the operations in an LTL terminal that are not adequately addressed in the literature. These characteristics include:

1. *Truck Arrival Times*: Arrival times of trucks do not follow a precise schedule. The drivers may call ahead and estimate the arrival time, however traffic congestion and other contingencies prevent precise planning. In the LTL industry, the assignment of inbound trucks to strip docks is made after the truck arrives at its destination.

2. *Truck Departure Times*: Departure times associated with outbound trucks also exhibit considerable variability. For example, some trucks bound for end-of-line terminals must closely follow a time schedule to make sure that the freight at the destination terminal is available to be carried back to the hub. On the other hand, trucks that operate between breakbulk terminals have a time window for departure.

3. *Congestion*: A portion of the inbound freight is staged before it can be loaded onto its outbound trailer. The staged freight, and loading and unloading of freight at the docks lead to internal congestion that effect the efficiency of operations within the terminal [11].

4. *Destination Types*: A variety of constraints must be considered when handling inbound freight based on its destination type. For example the freight that arrives from end-of-line terminals that is destined for other breakbulk terminals is handled differently and in a different section of the warehouse than freight from other breakbulk terminals that must be sorted and delivered to local terminals. Furthermore a single trailer may contain freight destined to more than a single destination type [11].

5.*Freight Flow*: The LTL industry, unlike other transportation industries such as rail and air, does not always adhere to strict time schedules, the variability of type of freight is higher, and there are peak times during the day, the month and the year where the demand varies on the order of 10% - 20%. This variability requires robust resource allocation methods to accommodate the high degree of variability. Also, one very important difference of LTL industry from airline industry is that freight is incapable of moving by itself unlike passengers.

Proper assignment of incoming truck to docks may reduce the time freight spends in the terminal. In practice, supervisors try to assign incoming trailers to docks close to the destination trailers for which they have the most freight. All such assignments are constructed based on intuition, or perhaps with the help of some simple spreadsheet calculations. This becomes difficult when the supervisor must consider other issues like making an assignment as the terminal gets larger or managing priorities for shipments that require rapid turnaround. Ultimately, the supervisor's goal is to make assignments that minimize work and this almost always involves minimizing worker travel [12].

Research in dock assignment problem at freight terminals have concentrated on minimizing weighted distances [14]. However, an objective of trying to minimize weighted distances in LTL terminals can lead to congestion which can adversely increase average freight times inside the terminals [12].

1.2.2.2 Performance Metrics of a Terminal

To improve performance of a terminal one should clearly identify the performance metrics of that terminal. Some of these metrics are similar to the ones at conventional warehouses. Two common performance metrics of terminals are average days delayed and average cycle time. Average days delayed is the average number of days required to ship an item from its arrival to the warehouse and average cycle time is the difference between the time an item arrives at the warehouse and the time it is ready for shipping. Both of these metrics have flaws in accurately comparing performances of independent warehouses and both are internally-focused [15]. They tend to improve customer service, but what matters at the end of the day is whether the customer received the shipment any earlier or not.

Gue proposes that Percent making Cut-Off (PCO), a metric that records the fraction of orders arriving before an established cut-off time that make the next shipment cycle, would be invulnerable to above mentioned problems [15].

1.3 Motivation

Physical size of the transportation sector, and its important role in supporting other important industries such as manufacturing, agriculture and mining, foster the need for operations research within this sector to improve the efficiency of its operations and improve service quality.

According to a survey released in 2001 by Logistics Institute at Georgia Tech (TLI) [16], majority of the transportation companies are not utilizing modern technology despite growing complexity of transportation systems. Approximately two thirds of the respondents use manual processes for many operations and they indicate that customer service and efficiency/cost saving rank highest in importance among factors regarding transportation planning.

The reflection of inherent variability in LTL industry on truck arrival schedules and terminal operations cause diversions from planned service levels. LTL industry needs operations research tools to incorporate models that provide more robustness in the daily operations. The proposed model fulfills the critical need to provide robust dock assignments at LTL terminal docks by intelligently locating time buffers between discrete events in order to absorb stochasticity. A similar approach to airport gate assignment model has been reported to provide significant improvement [17].

1.4 Research Objective

Strip dock assignment at terminals is subject to significant uncertainty due to the variability in truck arrival and unloading times. The objective of this thesis is to develop a robust model for dock assignments at strip docks to accommodate the deviation from the scheduled arrival and unloading times. The service level is measured by PCO metric and the goal is to increase the fraction of the freight that makes the next outbound shipment. Different from existing approaches, the model does not incorporate a distance minimizing objective.

1.5 Thesis Organization

Chapter 2 reviews the previous work in literature concerning modeling of similar problems. Chapter 3 defines the proposed MIQP model and discusses several illustrative examples. Chapter 4 introduces a planning level dock assignment heuristic which can be used to solve realistic size problems that could not be solved by the optimal model. Also in Chapter 4, the dock assignment heuristic is benchmarked with existing optimal solutions fro small problems. In Chapter 5, the effectiveness of the dock assignment heuristic is evaluated considering the previously discussed variability in system parameters. Chapter 6 introduces a dynamic dock assignment heuristic that can be implemented in real-time. The dynamic heuristic is then compared with the planning level dock assignment heuristic. Chapter 7 is a case study that compares the dynamic assignment heuristic to static assignment models that exist in practice. Chapter 8 concludes this thesis with conclusions and future research directions.

CHAPTER 2
LITERATURE REVIEW

2.1 Combinatorial Optimization

In the past decade the problem of finding optimal solutions to problems that can be structured as a function of some decision variables in presence of some constraints has been widely studied. Such problems can generally be formulated as follows:

Minimize $f(x)$

Subject to $g_i(x) \geq b_i; \quad i=1,\ldots..,m$

$h_j(x) \geq c_i; \quad i=1,\ldots..,n$

Where *x* is a vector of decision variables, f (•), g_i(•) and h_j(•) are general functions and *b, c, m, n* are constants. The problem can easily be modified for a maximizing objective.

There are many specific classes of optimization problems, obtained by placing restrictions on the functions under consideration, and on the values that the decision variables can take. One of the most well-known of these classes is that obtained by restricting f (•), g_i(•) and h_j(•) to be linear functions of decision variables which are allowed to take fractional (continuous) variables. The general class of problems that fit this description is called *Linear Programming* (LP) [18].

Another class of problems is combinatorial problems. This term is usually used for the problems in which the decision variables are discrete, i.e. where the solution is a set, or a sequence, or integers or other discrete objects. C*ombinatorial optimization* is the science of decision making in presence of discrete alternatives [19]. Combinatorial problems typically involve findings groupings, orderings, or assignments. *Integer and combinatorial optimization* deals with problems of maximizing or minimizing a function of variables subject to inequality and equality constraints and integrality restrictions on some or all the variables. A remarkably rich variety of problems can be represented by discrete optimization models.

Some of the applications covered by integer and combinatorial optimization include operational problems such as distributions of goods, production scheduling, machine sequencing, planning problems such as capital budgeting, facility location, portfolio analysis, communication and transportation network design and the design of automated production systems [20]. Some more recent applications are found in artificial intelligence, bio-informatics and machine learning [21].

Some well-known instances of combinatorial optimization problems are the assignment problem, the 0-1 knapsack problem, the set covering problem, the vehicle routing problem and the traveling salesman problem.

2.2 The Assignment Problem

The assignment problem, also known as the matching problem, is a classical and important combinatorial optimization problem. Assignment problems deal with the question how to assign a number of items (e.g. jobs) to a number of locations (e.g. workers). They consist of two components: the assignment as underlying combinatorial structure and an objective function. Mathematically an assignment is nothing else than a bijective mapping of a finite set into itself, i.e. permutation. Every permutation Φ of the set $N = \{1, …., n\}$ corresponds in a unique way to a permutation matrix $X \Phi = (xij)$ with $xij = 1$ for $j = \Phi(i)$ and $xij = 0$ for $j \neq \Phi(i)$. Assignments can be represented in various ways as depicted in Figure 5.

$$\Phi = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 4 & 1 \end{pmatrix}$$

$$X_\Phi = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

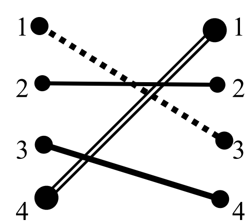

Figure 5 Different Representations of Assignments [22]

For the general formulation of assignment problems, a 0-1 integer variable is introduced, denoted below by y. One common use of 0-1 variable is to represent binary choice. Consider an event that may or may not occur, and suppose that it is part of the problem to decide between these possibilities. Let,

$$y = \begin{cases} 1 & \text{if the event occurs} \\ 0 & \text{if the event does not occur} \end{cases}$$

Assume that there are n people and m jobs and that each job must be done by exactly one person; also, each person can do at most one job. The cost of person j doing job i is c_{ij}. The problem is to assign the people to the jobs in a way that the cost of completing all the jobs is minimized. If ith job is assigned to jth person then $y_{ij} = 1$, if not $y_{ij} = 0$. The problem formulation would then be as follows:

$$Min \quad Z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} y_{ij} \qquad \text{(Minimize the cost of sum of all assignments)}$$

$$\sum_{j=1}^{n} y_{ij} = 1, \quad i = 1,2,\ldots.,m \qquad \text{(Job } i \text{ can be done by exactly one person)}$$

$$\sum_{j=1}^{m} y_{ij} \leq 1, \quad i = 1,2,\ldots.,n \qquad \text{(Person } j \text{ can do at most one job)}$$

The class of similar problems is referred to as *Linear Assignment Problems* (LAP), where the decision variables are linearly restricted with the objective and constraining functions.

2.3 Quadratic Assignment Problem (QAP)

Another class of assignment problems is the Quadratic Assignment Problem (QAP). The QAP covers a broad class of problems that involve the minimization of a total pair-wise interaction cost among a number of facilities.

The QAP is similar to a linear assignment problem but it has a quadratic component, a quadratic objective function. For example, facility location problem is a commonly addressed QAP where, given a set of n locations and n facilities, the objective is to find an assignment of all facilities to all locations such that the total cost of the assignment is minimized [23].The cost of each possible assignment is determined by the traffic intensity, commonly referred to as flow, between each pair of facilities and the distance between the assigned locations. The overall cost is the addition of all individual costs. For example, say four facilities were assigned to four locations as depicted in Figure 7.

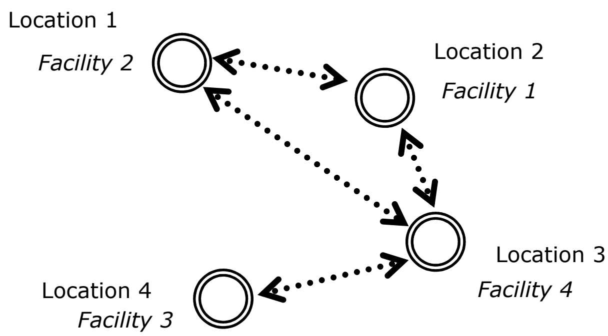

Figure 6 Assignment of Facilities to Locations [24]

The distances are denoted with d(•,•) and flows with f(•,•); e.g. d(1,2) and f(1,2) indicating the distance between locations 1 and 2 and flow between facilities 1 and 2. The cost of the assignment shown in Figure 6 would then be:

Cost (Figure 6) = d(1,2) * f(1,2) + d(1,3) * f(2,4) + d(2,3) * f(1,4) + d(3,4) * f(3,4)

The cost of the alternative assignment shown in Figure 7 would be:

Cost (Figure 7) = d(1,2) * f(3,4) + d(1,3) * f(1,4) + d(2,3) * f(1,3) + d(3,4) * f(1,2)

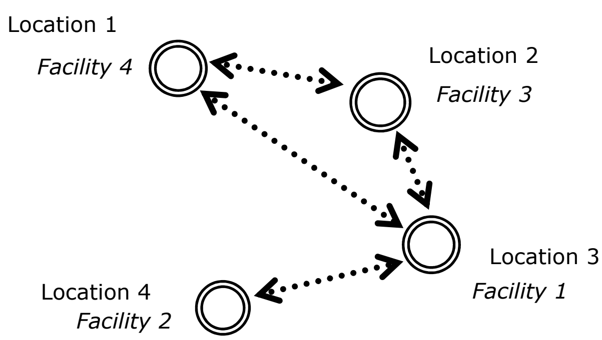

Figure 7 Alternative Assignment

The assignment that minimizes the cost would be the optimal solution, for the above example, if Cost (Figure 6) is less than Cost (Figure 7) then optimal assignment is Figure 6.

Quadratic assignment problems model many applications in diverse areas in operations research and combinatorial data analysis. In addition to its application in facility location problems, the QAP has been found useful in applications such as scheduling [25], the backboard wiring problem in electronics [26], parallel and distributed computing [27], and statistical data analysis [28]. Other applications can be found in [29-31]. An extensive survey on the formulations and solution methodologies of QAP can be found in [23].

One of the most distinguished instances of QAP is the *Gate Assignment Problem* (GAP). The total passenger walking distance in an airport is based on the passenger transfer volume between every pair of aircrafts and the distance between every pair of gates. Therefore, the problem of assigning gates to arriving and departing flights at an airport is a quadratic assignment problem, commonly formulated as a 0-1 integer problem [32]. The dock assignment models represented in the context of this thesis are motivated by the GAP, therefore the following section gives insight to this area.

2.4 The Gate Assignment Problem (GAP)

The GAP is an easily understood but difficult to solve problem. Integer programming and simulation have been applied to the GAP models in the modeling stage. The most occurring 0-1 integer representation of QAP in literature is the formulation of Koopmans and Berkmann, represented below in the original notation with facilities and locations [33].

$$Min \qquad Z = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{l=1}^{N} f_{ik}c_{jl}x_{ij}x_{kl} \qquad \text{(Eq. 1)}$$

Subject to:

$$\sum_{j=1}^{N} x_{ij} = 1, \quad i = 1,2,....,N \qquad \text{(Eq. 2)}$$

$$\sum_{i=1}^{N} x_{ij} = 1, \quad j = 1,2,....,N \qquad \text{(Eq. 3)}$$

$$x_{ij} \in \{0,1\} \quad i, j = 1,2,.....,N \qquad \text{(Eq. 4)}$$

where:

N = total number of facilities / locations

a_{ij} = fixed cost of locating facility i at location j

f_{ik} = flow of material from facility i to facility k

c_{jl} = cost of transferring a material unit from location j to location l

x_{ij} = 1, if facility i is at location j; 0 otherwise

The cost associated with placing a facility (i.e. aircrafts) at a certain location (i.e. gates) depends on not only the distance from other facilities and demands, but also on the interaction with other facilities [32]. Equation (2) ensures that every facility is assigned to a location, Equation (3) ensures that one location is assigned exactly one facility and Equation (4) indicates that variable x belongs to binary class. The constraints may be modified in accordance to the restrictions depicted by the problem statement.

The main purpose of airport GAP is to find the optimal gate-flight assignments to provide the most convenient boarding/deplaning operations to increase operating efficiency of the airport, i.e. increase the number of passengers who make their connecting flights. For example, Equation 1 in above model minimizes the flow times cost vector where the flow would represent the number of passengers and cost would represent the distance between connecting airplanes. Efficient assignments play an important role in alleviating the airport congestion which has become a profound problem with today's airline passenger volumes.

The complex process of gate assignment usually takes into account the following factors [34]:

1. Walking distances for transfer, terminating and originating passengers
2. Baggage handling distance for transfer, terminating, and originating passengers
3. Time tables of flight schedules
4. Aircraft gate size compatibility

Several models have been developed to represent airport gate assignments. Most of these models are formulated as 0-1 integer or mixed integer (linear or quadratic) programs. *Mixed integer quadratic problems* (MIQP) are 0-1 integer quadratic problems also including continuous variables. Objective functions of these programs usually aim to minimize the total passenger walking distance, the number of off gate events, the range of unutilized time periods for gates or a combination of these. The two most common constraints used in the modeling are,

1. Every flight must be assigned to a gate
2. No two flights can be assigned to the same gate concurrently

Braaksma [35] demonstrated that that the procedure of minimizing walking distances can have a significant impact on passenger walking distances without changing the layout of the terminal area. Babic et al. [36] formulated the GAP as a linear 0-1 integer program, not considering the transferring passengers. The optimal strategy reported by this paper is that aircraft carrying more passengers should be assigned to the gate closer to the central part of the building. Similarly, Mangoubi and Mathaisel [37] used linear programming relaxation of an IP formulation including the transfer passengers. The definitions of distances are defined similarly in these three studies.

Bihr [38] used 0-1 LP to solve the minimum walking distance GAP for fixed time intervals in a hub operation. The criterion selected in this paper is minimization of total passenger distance travel for a given arrival departure cycle. His model is a modification of the Koopmans and Berkmann 0-1 integer model. The first part of equation (1) is neglected, since there are no fixed costs associated with making assignments. The result is one of the most referred 0-1 integer GAP models:

$$Min \qquad Z = \sum_{i=1}^{m} \sum_{j=1}^{k} c_{ij} x_{ij} \qquad i = 1,....,m \ \ and\, j = 1,....,k$$

Subject to

$$C_{ij} = \text{PAX}\,(\,i, j\,) * \text{DIST}\,(\,i, j\,) \qquad i = 1,....,m \ \ and\, j = 1,....,k$$

$$\sum_{j=1}^{N} x_{ij} = 1\,, \qquad i = 1,2,....,N$$

$$\sum_{i=1}^{N} x_{ij} = 1\,, \qquad j = 1,2,....,N$$

$$X_{i,j} = 0 \text{ or } 1$$

Where:

l = number of arrival gates, k = number of departure gates, m = number of arrival flights

PAX (i, j) = number of passengers arriving on flight i and departing from gate j

$i = 1,\ldots,m$ and $j = 1,\ldots,k$

DIST (i, j) = number of passenger-distance units from gate i to gate j

$i = 1,\ldots,l$ and $j = 1,\ldots,k$

The above formulation is in 0-1 integer form and constitutes the foundation of many GAP models. All four papers report improvements in the walking distances. Baron [39] used simulation analysis to analyze the effects of different gate assignments on passenger walking distances.

Haghani [32] introduced the multi-slot (several ground time) gate assignment problem, extending the quadratic assignment problem formulations by incorporating time constraints (using time tables and flights schedules). It is reported that using multi-slot time intervals versus single-slot (one time period) or a batch of aircraft, allows effective gate utilizations. The flights are assigned to same gates if their time schedules are not conflicting. The study associates the several ground time periods with the decision variables such that:

$$X_{ijt} = \begin{cases} 1 & \text{if flight } i \text{ is assigned to gate } j \text{ in time period } t \\ 0 & \text{otherwise} \end{cases}$$

Other models that incorporate time into the GAP can be found in [40-42].

All the above models that were reported to be good in theory, however, many do not evaluate stochastic flight behaviors that usually occur in real life. Gosling [43] and Srihari and Muthukrishnan [44] applied expert systems to GAP and reported that these systems offer some advantages over conventional gate assignments in dealing with uncertainty. Most of the research in the area of GAP has concentrated on the static gate assignments. Static assignments are usually projected over a certain period of time, using historical data from a certain period of time. For example, using two months of historical data, one can project the gate assignment over the upcoming month. This model however, would be susceptible to the variability caused by internal or external factors, e.g. flight delays or monthly flight patterns. Real time assignments are done usually for much shorter time intervals, ideally just before the plane needs to be assigned a gate. The interrelationship between static gate assignments and real time assignments is affected by the variability in flights schedules (e.g. delays). Without considering flight delays in gate assignment, the model may have problems finding feasible solutions if there are more flights than gates in a time slot (e.g. peak hours and many delays). Some models have suggested ways to help resolve the issue with stochastic flight delays in the planning stages. For example, Hassounah and Steuart [45] showed that planned scheduled buffer times could improve schedule variability. Yan and Chang [46] and Yan and Huo [47] used fixed buffer times between continuous flights at the same gate in the static model, to help absorb the stochastic flight delays.

Some multi-objective models also have been studied. The model of Yan and Huo [47] considers the minimization of both the passenger walking distance and their waiting time. The paper reports usefulness for actual operations.

2.5 Solution Methodologies for QAP

As suggested by Sahni and Gonzales [48] QAP is an NP hard problem. It is still considered a computationally challenging task to solve modest size problems.

Relaxation by linearization is an efficient pre-solving technique by which QAP can be relaxed to a 0-1 linear integer problem. This technique is called *Reformulation Linearization Technique* (RLT). Since it is computationally less difficult to solve linear problems than quadratic ones, RLT technique is reported to be useful. For example, a new decision variable such as Y_{ijkl} can be introduced to represent $X_{ij} * X_{kl}$ (each possible pair of flight-to-gate assignment) allowing the linearization of the quadratic objective function. Some papers that explain RLT in QAP can be found in [23, 49]. A GAP model that uses this technique is in [32].

A diverse and vast range of complete solution methodologies have been applied to the QAP. Solution methodologies can be grouped into following two categories:

1. Optimal (Exact) Algorithms
2. Suboptimal (Heuristic) Algorithms

QAPLIB [50] is an online library that consists of many quadratic assignment problem instances along with a list of current known feasible solutions. It gives the researchers an opportunity to benchmark solutions for quality and computational performance.

Suggested solution algorithms in the literature, especially GAP literature, usually focus on suboptimal algorithms that prove to be computationally faster.

2.5.1 Optimal (Exact) Algorithms

Some of the exact algorithms that are well studied in QAP literature are dynamic programming, cutting plane and branch-and-bound techniques.

Dynamic programming approach has been first used by Christofides and Benavent [51] and cutting plane methods were introduced by Bazaraa and Sherali [52]. The algorithms are reported to obtain good results but are rather computationally time consuming.

Branch and bound is the most effective technique for problems of modest size. Among the types of branch-and-bound, single assignment algorithms are known to perform better [23]. The QAP literature utilizing the Branch and Bound technique is plentiful. Some of this work includes; parallel algorithms for the QAP [53, 54], lower bound calculation [55] , lower bound calculation and lagrangian relaxation [56], an accelerated branch and bound algorithm [57], special QAP cases [58-60], lower bound calculation using interior point calculation [61], feeding tree method to accelerate branch and bound [62] and Hungarian method [63].

Majority of work in GAP solution methodology literature comprises heuristic designs, since computation time is an important issue in both static and real time gate assignments. Babic at al. [36] and Mangoubi and Mathaisel [37] use LP relaxations to obtain optimal solutions and customized branch and bound type algorithms to compare quality. Bihr [38] uses LP relaxation and Hungarian method and reports the need for more sophisticated heuristics.

2.5.2 Sub-optimal (Heuristic) Algorithms

Heuristics are methods that maintain a trade off balance between the computational performance and quality. The quality of the solution is its nearness to the optimal solution obtained by exact algorithms mentioned in the previous chapter. There is intensive research in the heuristics design aiming to achieve higher quality solutions and better computational times. Frequently mentioned methods in current QAP literature are reviewed below.

2.5.2.1 Local and Tabu Search

Both Local search and Tabu search use initial solutions and move to a better solution, thus are called improvement algorithms. Both are iterative methods which terminate when there's no better solution in the neighborhood. Tabu Search; introduced by Glover [64], is a technique that additionally limits the search directions for each step while preventing local optimality. For example, items placed on the "tabu list" will not be re-computed, resulting in improvements in the computational performance. A study on the quality of local search for QAP can be found in [65] and a robust method for tabu search is studied in [66].

Some GAP related work includes; tabu search algorithm [41], and tabu search and memetic algorithms [42].

2.5.2.2 Simulated Annealing

Simulated annealing benefits from the analogy between combinatorial optimization and statistical mechanics. Simulated annealing also overcomes local optimality and is reported to be useful [67]. Some work in QAP literature include using the Tabu search technique for QAP [68], an extended study [69], an improved annealing scheme [70], optimization methods by simulated annealing [71] and a GAP related study [40].

2.5.2.3 Genetic Algorithms

Major advantages of genetic algorithms are that they are anytime algorithms which means that they can be interrupted anytime and will always have a result available, and that they are inherently parallel which is useful for solving quadratic assignment problems with the parallel computational capabilities of today's computers. Studies in genetic algorithms and QAP include the genetic approach to QAP [72], a greedy genetic algorithm [73] and a GAP model [74].

2.5.2.4 Other Algorithms

Some other algorithms are; network models for QAP [75], solving QAP with clues from nature [76], GRASP with path-relinking [77] and ant colonies for QAP [78]. GAP related work include network models [46], critical path method [79], a neural network design [80] and generalized heuristics [81].

Some literature that includes the comparison of heuristic methods can be found in [82], comparison of local search heuristics in[83], comparison of evolutionary heuristics in [84] and the very first heuristics used in solving QAP [85].

2.6 Dock Assignment Problem (DAP) Literature

Dock assignment problem, like the GAP, is a quadratic assignment problem and is combinatorial by nature. Although extensive research has been applied to the area of GAP, there is only a few studies directly related to the DAP. This thesis benefits from the research conducted in the GAP area, therefore GAP literature is reviewed in the previous sections. Research directly related to DAP is reviewed in this section.

Tsui and Chang [14] formulated the dock assignment problem using Koopmans and Berkmann's 0-1 integer representation with the objective of minimizing the total distance traveled to move items. It is assumed that there are M receiving docks, N shipping docks, I origins and J destinations for the terminal. There is no fixed cost associated with making assignments to docks. Let X_{im} = 1 if origin is assigned to receiving dock m, X_{im} = 0 otherwise and Y_{in} = 1 if destination j is assigned to shipping dock n, Y_{in} = 0 otherwise. Let W_{ij} represent the number of forklift trips required to move items that originate from i to destination j and let d_{jl} represent the distance between dock j and l.

$$Min \qquad Z = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N} w_{ij} d_{mn} x_{im} y_{jn} \qquad \text{(Eq. 5)}$$

Subject to:

$$\sum_{m=1}^{M} x_{im} = 1, \qquad i = 1,\ldots,I \qquad \text{(Eq. 6)}$$

$$\sum_{i=1}^{I} x_{im} = 1, \qquad m = 1,\ldots,M \qquad \text{(Eq. 7)}$$

$$\sum_{n=1}^{N} y_{jn} = 1, \qquad j = 1,\ldots,J \qquad \text{(Eq. 8)}$$

$$\sum_{j=1}^{J} y_{jn} = 1, \qquad n = 1,\ldots,N \qquad \text{(Eq. 9)}$$

$x_{im} \in \{0,1\} \qquad for\ \forall\ i$

$y_{jn} \in \{0,1\} \qquad for\ \forall\ j$

Equation (6) ensures every origin is assigned only one receiving dock, (7) ensures each receiving dock is assigned only one origin, (8) ensures each destination is assigned a shipping dock and (9) ensures that each shipping dock is only assigned one destination.

The formulation above is a static distance minimization formulation and is one of the scarce DAP models is the literature. The model however, is faced with the same problem caused by the stochasticity of trailer schedules much like the examples in GAP literature. It is reported in the paper that changing shipping patterns only occasionally warrant re-adjustment of dock assignments. This may not be the case for today's highly fluctuating shipping patterns. As a solution methodology, an LP relaxation and Branch and bound technique are used and computational difficulties are reported. Cole et al. [86] solve a similar problem with the objective of minimizing total weighted distance using a genetic algorithm and report quality solutions and good computational times.

Gue and Bartholdi [11] suggest that a DAP model with the objective function of minimizing distance only does not necessarily improve the performance of the terminal since it can lead to internal congestion. A model of travel cost inside the terminal is described along with three types of congestion. Using these models alternative layouts are constructed. They report 11% improvement in productivity in a terminal. Gue [12] develop a look-ahead dock assignment model and report that the model cuts the labor costs by 15% compared to FCFS policy generally used by terminal supervisors.

CHAPTER 3
PROBLEM DESCRIPTION and MIQP SOLUTION

The solution approach to increasing robustness in freight transportation networks stems from the idea that an assignment which provides an even distribution of idle times at strip docks will tend to absorb the stochastic variability in the arrival and service schedules. The proposed model minimizes the variance associated with the distribution of the idle times of the docks at an LTL terminal.

The problem is formulated as a mixed integer quadratic problem (MIQP), it involves continuous and binary variables, and the objective function is quadratic. The formulation given in Section 3.2 will be referred to as the *optimal model* in the context of this thesis.

3.1 Problem Notation

The following notation will be used in the following sections:

$T_{Earliest,k}$	Earliest available time of dock k in the beginning of planning horizon
$T_{Latest,k}$	Latest available time of dock k at the end of planning horizon
A_j	Scheduled arrival of trailer j
G_j	Scheduled Service time (unloading or loading) of trailer j
$E_{j,k}$	Entering time of trailer j to dock k
$L_{j,k}$	Leaving time of trailer j from dock k
$S_{j,k}$	Slack (idle) time of trailer j at dock k
$S_{Last,k}$	Last Slack (idle) time of dock k after the last departure from this dock
N	Number of trailers
M	Number of docks
H	A very positive large integer
X_{jk}	1 if trailer j is assigned to dock k 0 otherwise

3.1.1 Earliest ($T_{Earliest,k}$) and Latest ($T_{Latest,k}$) Available Times of Dock k

Earliest available time ($T_{Earliest,k}$) indicates the beginning of the time windows for dock k, i.e. start of the planning window. Latest available time ($T_{Latest,k}$) indicates the end of the operation window. Dock availability of three docks has been exemplified in Figure 8 as timelines.

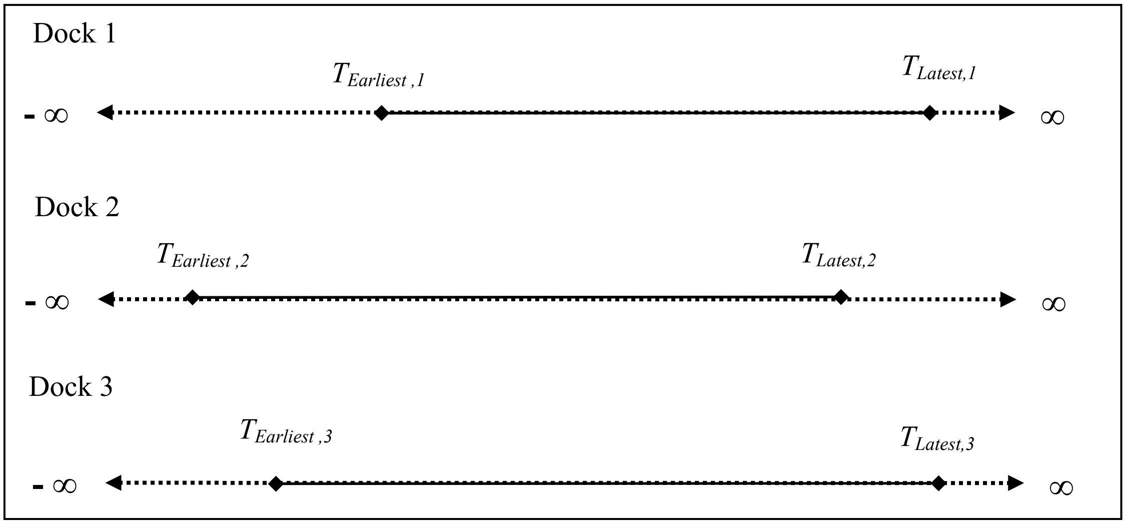

Figure 8 Timelines of Terminal Docks 1, 2 And 3

3.1.2 Scheduled Arrival Time of Trailer j (A_j)

Scheduled arrival time (A_j) is the time that is scheduled for a trailer to enter the terminal yard, later to be assigned to a dock. It is assumed that trailers are sorted in descending order, i.e. if $i > j$ then $A_i > A_j$. Figure 9 shows the timelines of trailers 1 and 4 as an example.

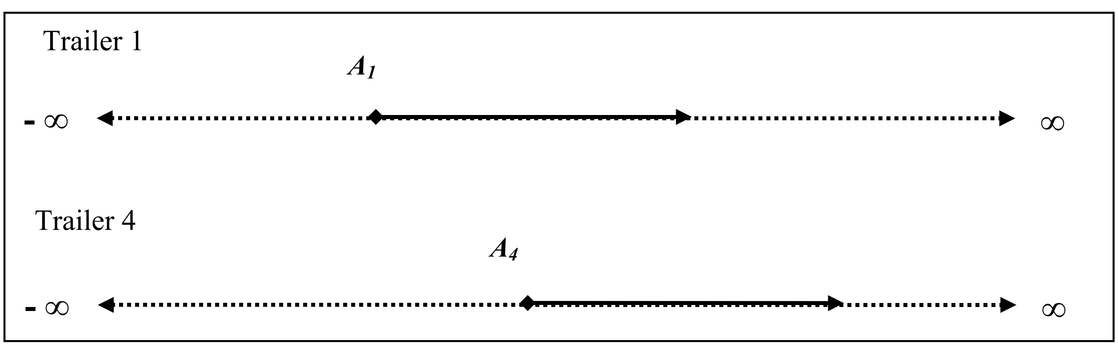

Figure 9 Timeline of Trailer 1 And 4

3.1.3 Enter ($E_{j,k}$) , Leave ($L_{j,k}$) and Service (G_j) Times of Trailer *j* on Dock *k*

Trailer *j* enters dock *k* at the assigned time denoted by E_{jk} . This time has to be equal to or greater than the scheduled arrival time of trailer *j* (A_j), and greater than the earliest available time of dock *k*. After G_j time units, when the trailer is serviced and leaves the dock, its leave time L_{jk} is assigned which must be equal or less than the latest available dock time ($T_{Latest,k}$) of dock *k*. Figure 10 shows the assignment of trailer 1 to dock 1 and trailer 4 to dock 2.

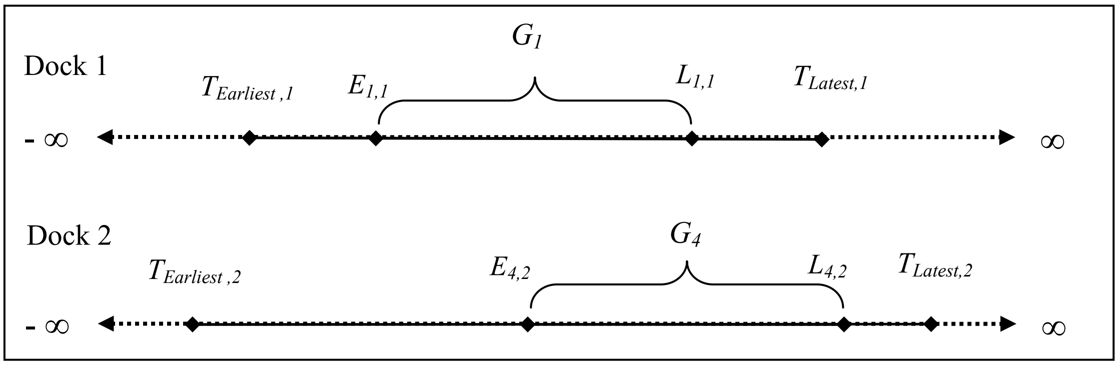

Figure 10 Entering and Leaving Times of Trailers 1 And 2

3.1.4 Slack Time of Trailer *j* on Dock *k* (S_{jk})

Slack time is defined as the idle time between the departure of the last trailer at a gate and the entering time of the current trailer. As depicted in Figure 11, trailers 3 and 9 are assigned to dock 6 consecutively. If a trailer is the first trailer assigned to a dock; i.e. no other trailers precede it, its slack time is its enter time minus the first available time of the dock, e.g. trailer 3 in Figure 11. If there is an assignment before a trailer, then the slack time is its enter time minus the leave time of the preceding trailer, e.g. trailer 9 in Figure 11. The last slack time at a dock ($S_{Last,k}$) is its latest available time ($T_{Latest,k}$) minus the leave time of the latest trailer assigned to the dock.

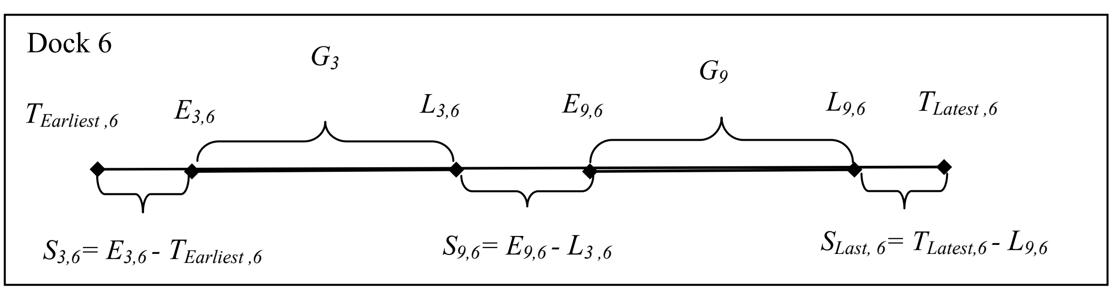

Figure 11 Slacks Due To Assignments of Trailers 3 And 9 On Dock 6

3.2 Problem Formulation (Optimal Model)

Min $$\boxed{\sum_{k=1}^{M}\sum_{j=1}^{N+1} S_{j,k}^{\;2}}$$ (Eq. 10)

Subject to:

$$\boxed{\sum_{k=1}^{M} x_{j,k} = 1} \qquad j = 1,\ldots,N$$ (Eq. 11)

$$\boxed{E_{j,k} \geq A_j X_{j,k}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 12)

$$\boxed{L_{j,k} - E_{j,k} = G_j X_{j,k}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 13)

$$\boxed{E_{j,k} \geq L_{i,k} + H(X_{j,k} + X_{i,k} - 2)}$$

$j = 1,\ldots,N \qquad k = 1,\ldots,M \qquad i = 1,\ldots,N$ *where* $i < j$ (Eq. 14)

$$\boxed{L_{j,k} \leq T_{Latest,k}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 15)

$$\boxed{2 * Y_{i,j,k} \leq X_{j,k} + X_{i,k}}$$

$j = 1,\ldots,N \qquad k = 1,\ldots,M \qquad i = 1,\ldots,N$ *where* $i < j$ (Eq. 16)

$$\boxed{\sum_{i=1}^{j-1} Y_{i,j,k} = 1} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 17)

$$\boxed{S_{j,k} \geq E_{j,k} - L_{i,k} + H(Y_{i,j,k} - 1)}$$

$j = 1,\ldots,N \qquad k = 1,\ldots,M \qquad i = 1,\ldots,N$ *where* $i < j$ (Eq. 18)

$$\boxed{X_{j,k} \in \{0,1\}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 19)

$$\boxed{Y_{i,j,k} \in \{0,1\}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 20)

$$\boxed{E_{j,k}, L_{j,k}, S_{j,k}, A_j, G_j, T_{Latest,k} \geq 0} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$ (Eq. 21)

3.2.1 The Objective Function

The objective function aims to spread the slack times as evenly as possible in order to create buffers to absorb variability in scheduled arrival (A_{jk}) and scheduled service times (G_{jk}). This is achieved by minimizing the square of the slack time (S_{jk}) (Eq.10) since this is equivalent to minimizing the variance.

$$Min \boxed{\sum_{k=1}^{M}\sum_{j=1}^{N+1} S_{j,k}^{\ 2}}$$

Theorem 1

Minimizing the sum of squares of slack times minimizes the variance of slack times.

Proof

The total slack time of dock s is the difference between the total available time of docks and the total service time of trailers. ($_{N+1}$ denotes the end-slack; $S_{Last,k}$)

$$\sum_{k=1}^{M}\sum_{j=1}^{N+1} S_{j,k} = \sum_{k=1}^{M} T_{Latest,k} - T_{Earliest,k} - \sum_{j=1}^{N} G_j$$

It is assumed that the total available time of the docks and the total service time of trailers are constant, thus the sum of idle times is constant, regardless of the way trailers are assigned.

For any g (x) and constants a and b, E[ag (x) + b] = a E[g (x)] + b

Thus;

Var (x) = E [(x – μ)^2] *(Var (x) is the second central moment at zero)*

= E [x^2 - 2 μ x – μ ^{2}]

= E [x^2] - 2 μ E [x] + μ ^{2} (*Property 1*)

= E [x^2] - μ ^{2}

Since μ = constant; minimizing E [x^2] minimizes Var (x) ▲

3.2.2 The Constraints

3.2.2.1 Constraint 1

$$\boxed{\sum_{k=1}^{M} x_{j,k} = 1} \qquad j = 1,\ldots,N$$

Constraint 1 guarantees that one trailer is assigned to only one dock.

3.2.2.2 Constraint 2

$$\boxed{E_{j,k} \geq A_j X_{j,k}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$

Constraint 2 ensures that the enter time of trailer j (E_{jk}) will be at least equal to or greater than its scheduled arrival time (A_j), if assigned. If not assigned, $X_{j,k} = 0$ and $E_{j,k} \geq 0$.

3.2.2.3 Constraint 3

$$\boxed{L_{j,k} - E_{j,k} = G_j X_{j,k}} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M$$

Constraint 3 states that the service time of a trailer (G_j) is its assigned enter time (E_{jk}) minus its leave time (L_{jk}), if assigned. If not assigned, $X_{j,k} = 0$ and $E_{j,k} = L_{j,k}$.

3.2.2.4 Constraint 4

$$\boxed{E_{j,k} \geq L_{i,k} + H(X_{j,k} + X_{i,k} - 2)} \qquad j = 1,\ldots,N \qquad k = 1,\ldots,M \qquad i = 1,\ldots,M \text{ where } i < j$$

Constraint 3 prevents time-conflicts (overlapping assignments). A trailer has to leave a dock in order for another one to be assigned. For example, trailer 3 is assigned a leave time of 9. If trailer 8 is assigned an enter time less than 9, then overlapping occurs, as depicted in Figure 12. The trailer can only be assigned to a time greater than or equal to 9 (assuming it does not violate any other constraints). Since $i < j$, X_{jk} denotes the trailer that is being assigned and X_{ik} denotes a search for the previously assigned trailers in the same dock (note that trailer times are sorted in descending order, i.e. if $i > j$ then $A_i > A_j$).

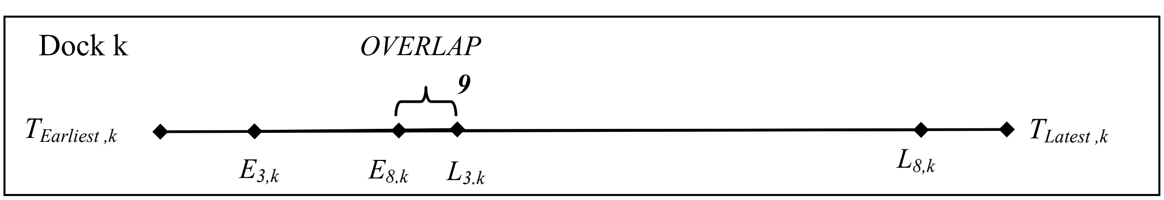

Figure 12 Overlapping of Trailer 8 on Trailer 3

If there is a previously assigned trailer in the dock; i.e. X_{ik} = 1 and the current assignment is also made; i.e. X_{jk} =1 ; then (X_{ik}.+ X_{jk}) = 2 and the RHS of constraint 4 will be equal to L_{ik} yielding $E_{j,k} \geq L_{i,k}$ which means that the considered assignment of enter time can only be made after the leave time of the preceding trailer.

If there is no previously assigned trailer in the dock, i.e. X_{ik} = 0 and the current assignment will be made or will not be made; i.e. X_{jk} =1 or X_{jk} =0; then (X_{ik}.+ X_{jk}) = 1 or (X_{ik}.+ X_{jk}) = 1 and the RHS of constraint 4 will be equal to a very large negative integer yielding $E_{j,k} \geq 0$ which is already in agreement with the negativity constraints.

3.2.2.5 Constraint 5

$\boxed{L_{j,k} \leq T_{Latest,k}}$ *j = 1,…,N* *k = 1,…,M*

This constraint ensures any leave time should be less than or equal to the latest available time of the gate.

3.2.2.6 Constraints 6, 7 and 8

These three constraints are used to determine slack times.

$\boxed{2 * Y_{i,j,k} \leq X_{j,k} + X_{i,k}}$ *j = 1,…,N* *k = 1,…,M* *i = 1,…,M where i < j*

Constraint 6 uses a similar logic to constraint 4.

Table 1 shows the four possible scenarios of constraint 6. If two trailers are not assigned consecutively, the value of Y will be zero. Otherwise it can be 0 or 1 (note that all variables are non-negative and binary).

Table 1 Possible Outcomes of Constraint 6

	$X_{i,k}$	$X_{j,k}$	$Y_{i,j,k}$	
Scenario 1	1	1	≤ 1.0	0 or 1
Scenario 2	0	1	≤ 0.5	0
Scenario 3	1	0	≤ 0.5	0
Scenario 4	0	0	≤ 0.5	0

$\boxed{\sum_{i=1}^{j-1} Y_{i,j,k} = 1}$ *j = 1,…,N* *k = 1,…,M*

Constraint 7 prevents an incorrect slack time calculation by forcing the model to choose a slack when there are more than 1 trailers assigned before a particular trailer (all the preceding trailers can be 0 or 1 and since it is a minimization problem, without Constraint 6 the model would make them all 0). Consider the example where trailer 3 is assigned to dock 1 along with trailers 1 and 2 as depicted in Figure 13. Constraint 9 then yields, $Y_{131}+Y_{231}=1$. One of the possible configurations of Y has to be 1; in this case, either Y_{131} *or* Y_{231}.

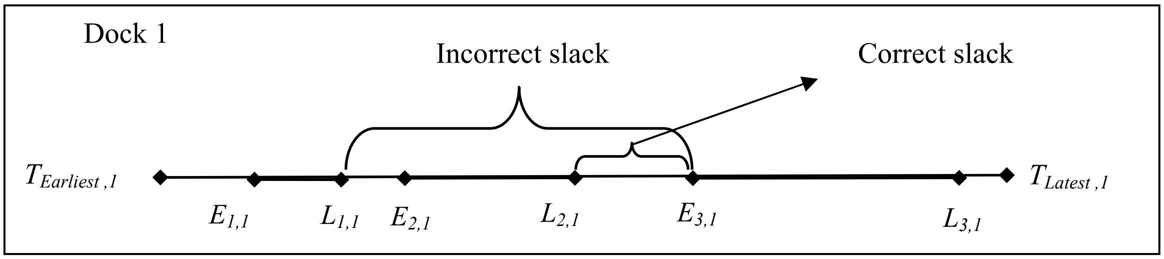

Figure 13 Finding Correct Slack with More Than One Preceding Trailers at the Same Dock

$$\boxed{S_{j,k} \geq E_{j,k} - L_{i,k} + Z(Y_{i,j,k} - 1)} \qquad j = 1,\dots,N \quad k = 1,\dots,M \quad i = 1,\dots,N \text{ where } i < j$$

Constraint 8 is used to determine the objective function value. Slacks are between two trailers, the latest available time of the dock and the latest assigned trailer at that dock, and the first assigned trailer and the earliest available time of that dock. As explained before, Y variable is an indication to whether or not these slacks have formed. Table 2 shows the possible outcomes of constraint 8 with all possible *Y* scenarios given in Table 1.

Table 2 Possible Outcomes of Constraint 8

	$X_{i,k}$	$X_{j,k}$	$Y_{i,j,k}$	$S_{j,k}$	Model will force
Scenario 1	1	1	0 or 1	≥ 0 or $\geq E_{jk} - L_{ik}$	$S_{jk} = E_{jk} - L_{ik}$
Scenario 2	0	1	0	≥ 0	$S_{jk}=0$
Scenario 3	1	0	0	≥ 0	$S_{jk}=0$
Scenario 4	0	0	0	≥ 0	$S_{jk}=0$

To illustrate the use of constraint 8, consider trailers 1, 2 and 3. The scenarios are:

1. If j^{th} trailer (Trailer 3) is immediately preceded by the $(j-1)^{th}$ trailer (Trailer 2). The algorithm searches and finds the assigned trailers and forces $S_{31} = E_{31} - L_{21}$, which is the correct decision.

2. If j^{th} trailer (Trailer 3) is immediately preceded by the $(j - x)^{th}$; where ($x < j - 1$) trailer (Trailer 1). The algorithm searches and finds the assigned trailers and forces $S_{31} = E_{31} - L_{11}$, which is the correct decision.
3. If j^{th} trailer (Trailer 3) is immediately preceded a trailer (Trailer 2) that is preceded by another trailer (Trailer1). The model has to satisfy constraint 7 and will force the lower of $Y_{231}$and Y_{131} to 1 (since it wants to minimize). It will always correctly choose $S_{31} = E_{31} - L_{21}$ over $S_{31} = E_{31} - L_{11}$ since the latter equation is always greater than the first one (minimizing objective).

   The scenarios and results are summarized in Table 3.

Table 3 Possible Outcomes of the Example

Condition	$X_{3,1}$	$X_{2,1}$	$X_{1,1}$	$Y_{2,3,1}$	$Y_{1,3,1}$	$Y_{1,3,1}+Y_{2,3,1}$	$S_{j,k}$	Model will force
2 precedes 3 ***2→ 3***	1	1	0	1	0	1	$S_{31} \geq E_{31} - L_{21}$ or $S_{31} \geq 0$	$S_{31} = E_{31} - L_{21}$
1 precedes 3 ***1→ 3***	1	0	1	0	1	1	$S_{31} \geq E_{31} - L_{11}$ or $S_{31} \geq 0$	$S_{31} = E_{31} - L_{11}$
1 precedes 2 precedes 3 ***1→ 2→ 3***	1	1	1	0 or 1	0 or 1	Model will force the lower of $Y_{231}$and Y_{131} to 1.	$S_{31} \geq E_{31} - L_{11}$ or $S_{31} \geq E_{31} - L_{21}$	$S_{31} = E_{31} - L_{21}$

3.2.2.7 Constraint 9 and 10

$\boxed{X_{j,k} \in \{0,1\}}$ $\quad j = 1,\ldots,N \quad k = 1,\ldots,M$

$\boxed{Y_{i,j,k} \in \{0,1\}}$ $\quad j = 1,\ldots,N \quad k = 1,\ldots,M \quad i = 1,\ldots,N$

These are the BINARY constraints. *X* and *Y* can be 0 or 1, indicating the assignment is a discrete event, it either occurs or it does not occur.

3.2.2.8 Constraint 11

$\boxed{E_{j,k}, L_{j,k}, S_{j,k}, A_j, G_j, T_{Latest,k} \geq 0}$ $\quad j = 1,\ldots,N \quad k = 1,\ldots,M$

This is the non-negativity constraint.

3.3 Illustrative Examples

Two examples of different size will be solved in this section. Example 1 has 4 trailers and 2 docks and Example 2 has 12 trailers and 4 docks. For Example 2 the software code is included in Appendix A.

3.3.1 Example 1

There are 4 trailers and 2 docks available between 0 and 10 unit time. The arriving schedule and estimated unloading times of the trailers are given in Table 4. The objective is to assign the trailers to the dock s in such a way that the slack times of the docks will evenly be distributed.

Table 4 Data for the Example 1

	Trailer 1	Trailer 2	Trailer 3	Trailer 4
Scheduled Arrival Time (*A*)	1	2	6	7
Unloading time (*G*)	4	4	3	3

3.3.1.1 An Arbitrary Feasible Solution to Example 1

A feasible solution to this problem is assigning trailers 1 and 4 to dock 1, and trailers 2 and 3 to dock 2. ($X_{11} = 1$, $X_{12} = 0$, $X_{21} = 0$, $X_{22} = 1$, $X_{31} = 0$, $X_{32} = 1$, $X_{41} = 1$, $X_{42} = 0$) The results according to this assignment are given in Table 6 and displayed in Figure 14.

Table 5 Feasible Assignments for the Example 1

	Trailer 1	Trailer 2	Trailer 3	Trailer 4
Dock assigned	1	2	2	1
Enter time to gate *E*	1 ($E_{1,1}$)	2 ($E_{2,2}$)	6 ($E_{3,2}$)	7 ($E_{4,1}$)
Leave time to gate *L*	5 ($L_{1,1}$)	6 ($L_{2,2}$)	9 ($L_{3,2}$)	10 ($L_{4,1}$)
Slack time	1 ($S_{1,1}$)	2 ($S_{2,2}$)	0 ($S_{3,2}$)	2 ($S_{4,1}$)

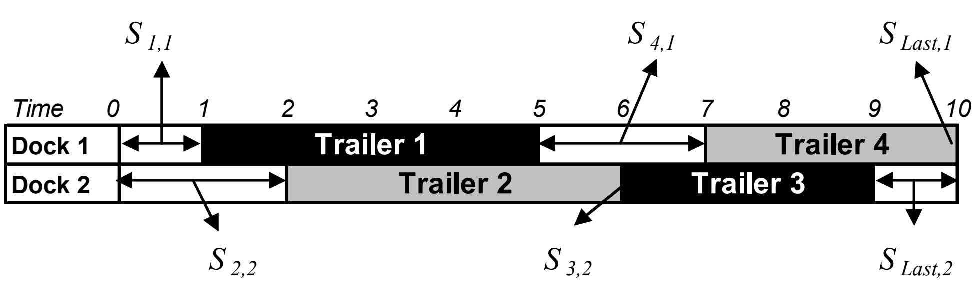

Figure 14 Representation of a Feasible Solution on Timeline

The objective function value of this assignment is:

$Z = (S_{1,1})^2 + (S_{4,1})^2 + (S_{2,2})^2 + (S_{3,2})^2 + (S_{Last,1})^2 + (S_{Last,2})^2$

$Z = 1^2 + 2^2 + 2^2 + 0^2 + 0^2 + 1^2$

$Z = 10$

3.3.1.2 Optimal Solution of Example 1

The problem is solved with the proposed model using commercially available *Xpress* [87] MIQP solver. The resulting assignments are given in Table 7.

Table 6 Assignment of the Proposed Model

	Trailer 1	Trailer 2	Trailer 3	Trailer 4
Dock assigned	1	2	1	2
Enter time to gate *E*	1 ($E_{1,1}$)	2 ($E_{2,2}$)	6 ($E_{3,2}$)	7 ($E_{4,1}$)
Leave time to gate *L*	5 ($L_{1,1}$)	6 ($L_{2,2}$)	9 ($L_{3,2}$)	10 ($L_{4,1}$)
Slack time	1 ($S_{1,1}$)	2 ($S_{2,2}$)	1 ($S_{3,1}$)	1 ($S_{4,2}$)

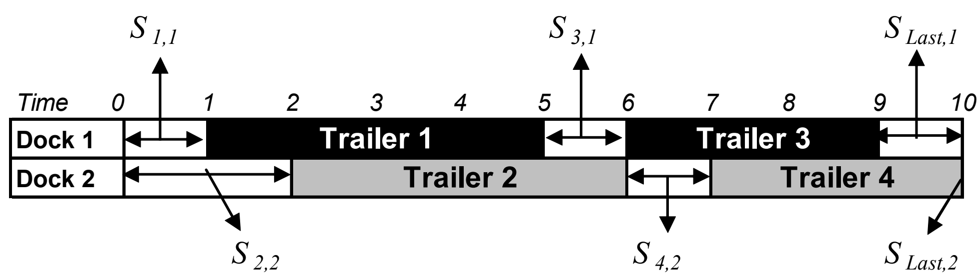

Figure 15 Representation of the Solution of Proposed Model

The objective function value of the robust assignment is:

$Z = (S_{1,1})^2 + (S_{3,1})^2 + (S_{2,2})^2 + (S_{4,2})^2 + (S_{Last,1})^2 + (S_{Last,2})^2$

$Z = 8$

The objective function value of the robust model (z = 8) is less than the objective function value of the initial solution (z = 10). As depicted in Figure 15, the proposed model distributes the slacks as evenly as possible. Trailer 3 now has a slack behind it that can absorb a possible extent in its service time, and trailer 4 now has a slack in front to absorb second trailer’s arrival time or service time variability.

3.3.2 Example 2

The code for Example 2 with 4 docks and 12 trailers is given in Appendix A. It should be noted that Xpress syntax does not allow 0 indices in vectors; e.g. *A(0, 0)*, therefore the indices of trailers start from 1, e.g. *S(2, 1)* represents the slack of first trailer at dock 1. The trailer indices range from 1 and to *NT+2* (where *NT* is the number of trailers). For example, in the case of the sample problem where there are 4 docks and 12 trailers, *S(13, 1)* represents the slack of the 12th trailer at dock 1 and *S(14, 1)* represents the latest slack at dock 1, previously named as $S_{Last,\ 1}$.

The proposed model is solved using Xpress-IVE module with 4 available docks and 12 trailers. The schedules of arriving trailers are shown in Table 7. For example, trailer 1 is scheduled to arrive at time 0 and is estimated to have a service time of 1 unit.

Table 7 Scheduled Arrival and Estimated Service Times for Example Problem 2

Trailer	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Scheduled Arrival Time (A)	0	1	1	1	3	3	4	5	6	8	8	9
Unloading time (G)	1	2	4	2	3	4	4	5	2	1	3	2

The value of the objective function which is obtained for this schedule by the optimal model is 14.25. The corresponding assignments are shown in Figure 16.

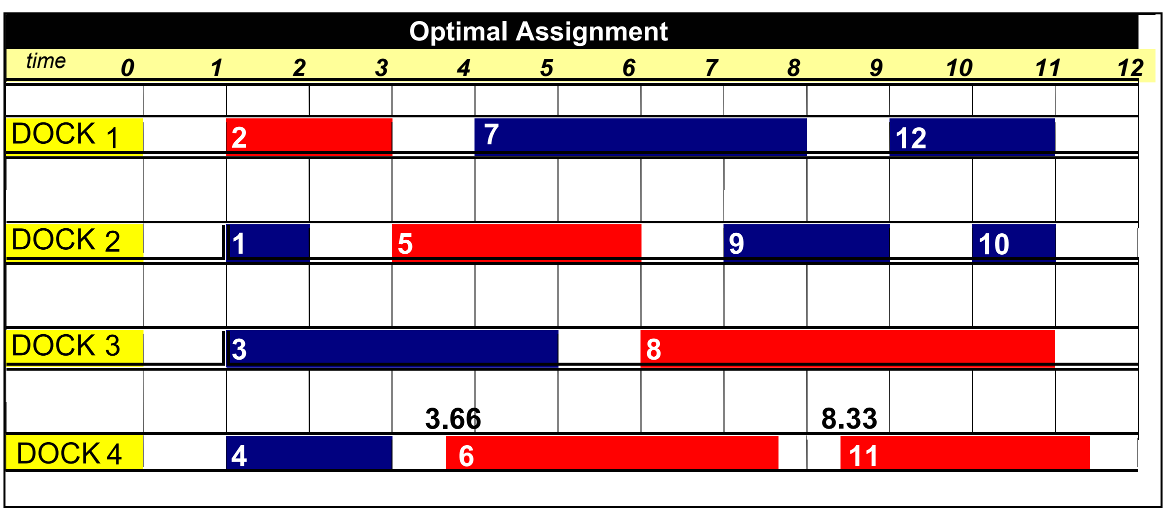

Figure 16 Graphical Representation of Robust Assignment for Sample Problem

3.4 Computational Issues

This section covers the computational aspects of the implementation by investigating the relationships between the number of variables/constraints and the problem size, and computational performance.

3.4.1 Problem Size

Due to the quadratic objective function and Y_{ijk} variable, the model generates a large number of variables and constraints which make the computation a non-trivial task. Table 8 displays the relationship between the number of trailers and number of model variables when number of docks is constant. A similar table is given in Table 9 with constant number of trailers and varying number of docks.

Table 8 Varying Number of Trailers

	2 Dock 4 Trailer	2 Dock 8 Trailer	2 Dock 12 Trailer	2 Dock 33 Trailer	2 Dock 67 Trailer
Constraints	92	278	558	3622	14194
Variables	58	146	264	1419	5159

Table 9 Varying Number of Docks

	2 Dock 4 Trailer	8 Dock 4 Trailer	12 Dock 4 Trailer	33 Dock 4 Trailer	67 Dock 4 Trailer
Constraints	92	372	556	1522	3086
Variables	58	248	372	1023	2077

Figure 17 and Figure 18 show the change in number of variables and constraints respectively, with the change in number of docks and trailers. The number of variables and constraints increase exponentially with increasing number of trailers and increase linearly with increasing number of docks.

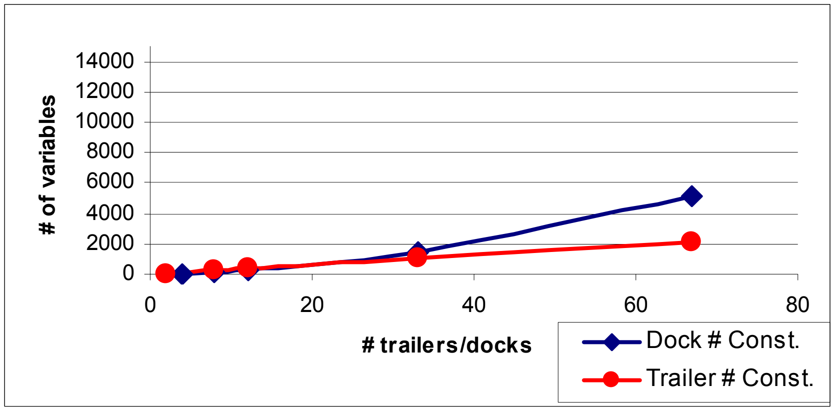

Figure 17 Effect of Increasing Trailers/Docks on the Number of Variables

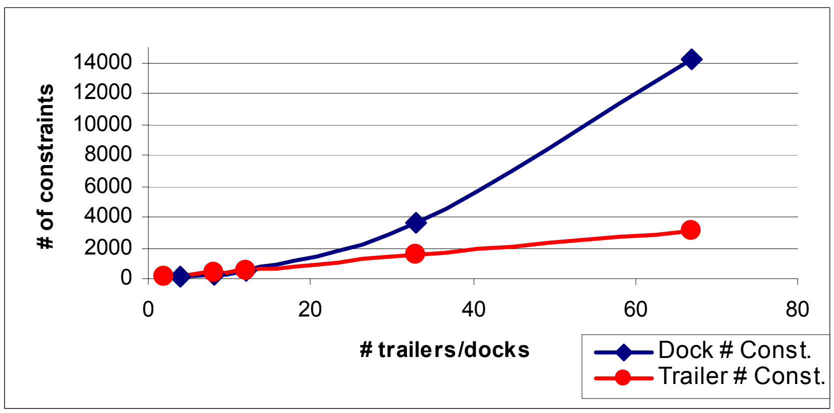

Figure 18 Effect of Increasing Trailers/Docks on the Number of Constraints

3.4.2 Computation Time

All experiments are run on a machine with Intel Pentium 4 @ 2.66 GHZ, 512 MB DDR Ram and Windows NT operating system. When the problem size is increased from 2-dock-4 trailer to 3-dock-8 trailer the computation time increases from 0.3 seconds to 2573.6 seconds.

An interesting observation about *Xpress* is that the software tends to yield good results in low percentile of the total computation time. For example, the case of 3 docks and 8 trailers yields its optimal solution in the 6% of the total run time. Furthermore, in only 1% of the total run time a good solution within 2.3% of the optimal solution is achieved as depicted in Figure 19. This is an indication that the model can be interrupted manually to reduce computational time with a trade-off in solution quality, however this has only been observed for small size problems and it is still a non-trivial task to solve even moderate size problems that exist in practice. The largest model that has reached an optimal solution in experiments is 4 docks-12 trailers at 129604 seconds.

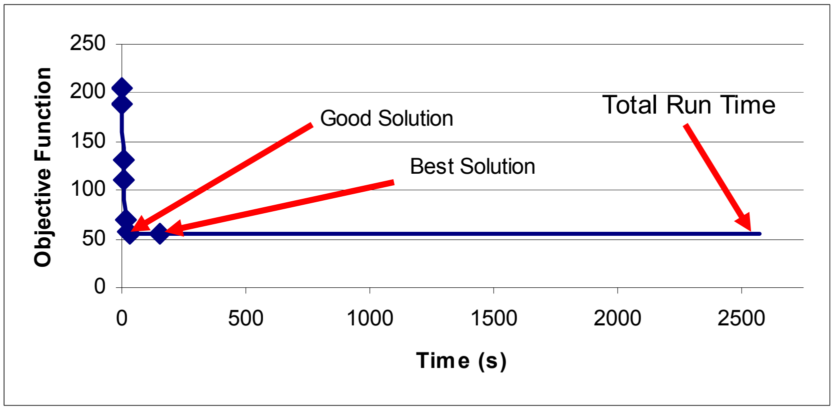

Figure 19 Graph Showing the Solution Achievement Timeline of 3 Dock 8 Trailer Problem

Computational experiments show that there is need for sub-optimal models for achieving solutions in polynomial time. The next chapter investigates the design of a dock assignment heuristic that produces sub-optimal solutions.

CHAPTER 4
DOCK ASSIGNMENT HEURISTIC

The computational difficulties associated with the optimal model require the development of a heuristic approach to solve practical size problems. A sub-optimal dock assignment algorithm which will be referred as the *Dock Assignment Heuristic* (DAH) is introduced in this chapter. DAH is then benchmarked with known optimal solutions for small size problems.

4.1 Heuristic Algorithm

The DAH minimizes the makespan on n parallel processors (docks) to maximize idle times at each dock which can then be evenly distributed between the trailers assigned on that dock. Example in Table 7 is revisited.

The steps of the DAH are as follows:

Step 1. Sort the trailers in ascending order of Estimated Arrival Times. (A_j)

Step 2. Assign the trailer to the dock with the smallest existing leave time (L_j) in sequential order as depicted in Figure 20.

Step 3. Assign the enter time to the trailer so that $E_{j,k} = A_j$. If A_j time slot is occupied by the previous assignment move it to the first available time and update its $E_{j,k}$ and $L_{j,k}$. ($L_{j,k} = E_{j,k} + G_{j,k}$). Repeat Step 3 for all trailers as depicted in Figure 21.

Step 4. Find the total idle time at dock k and distribute it evenly between the assigned trailers on dock k as depicted in Figure 22. (Note that assignments must not violate $E_{j,k}$ >= A_j.). Repeat Step 4 for all docks.

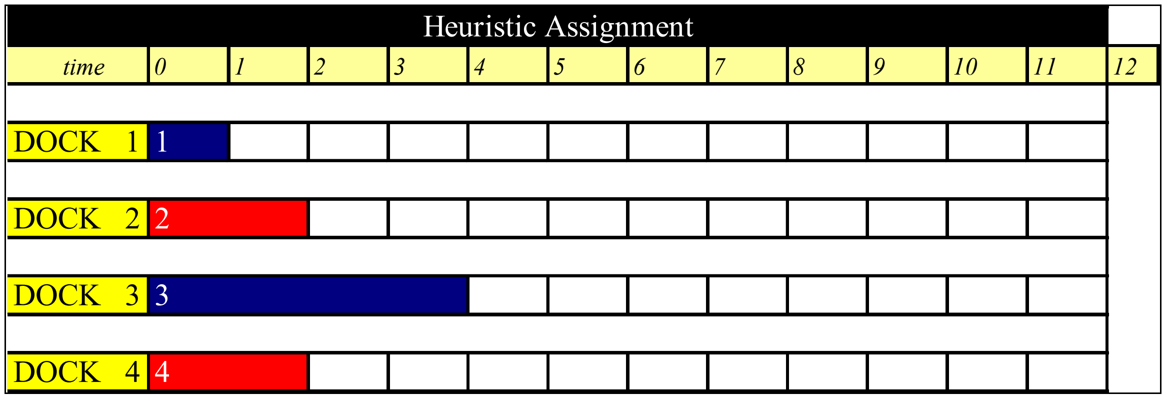

Figure 20 Step 2 of the Heuristic Algorithm Showing 4 Trailers

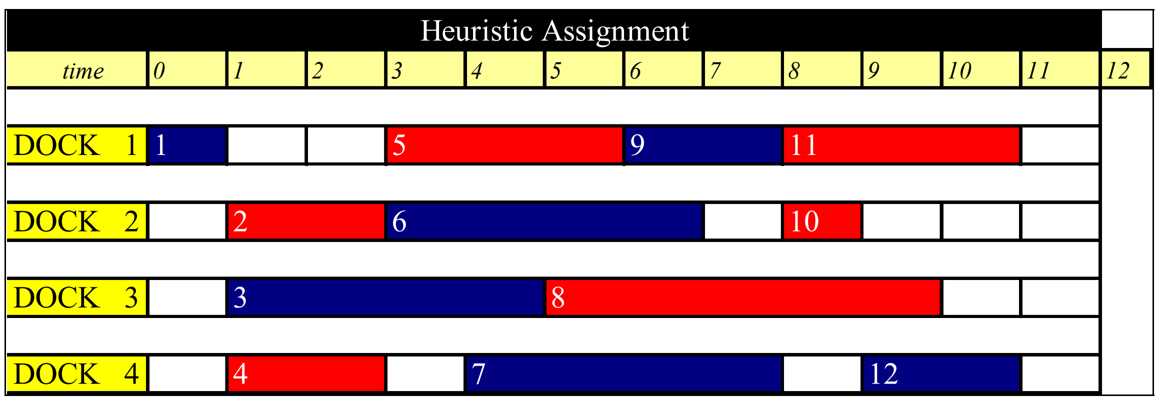

Figure 21 Step 3 of the Heuristic Algorithm

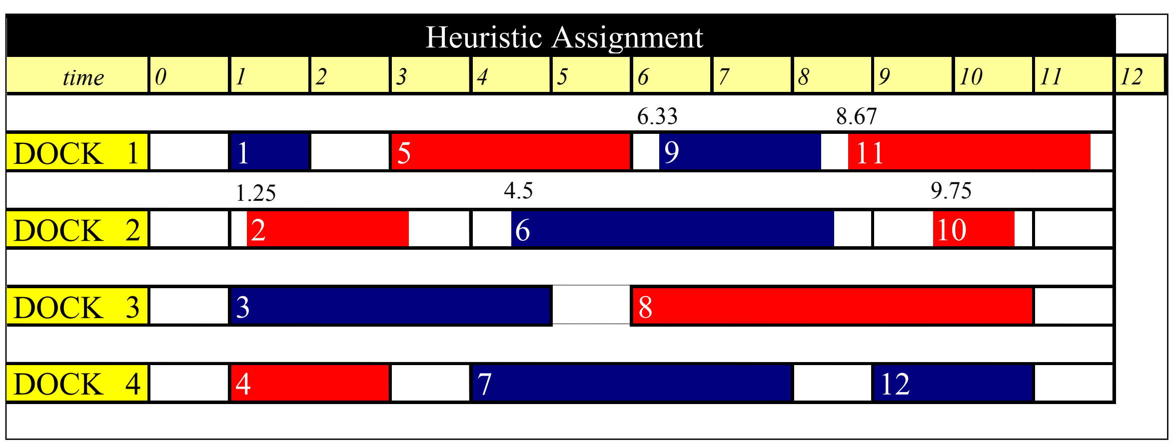

Figure 22 Step 4 of the Heuristic Algorithm

4.2 Effectiveness of the DAH

This section benchmarks the effectiveness of the heuristic by comparing it with the existing optimal solutions. 40 data sets for 9 trailers with varying dock numbers and dock utilizations have been randomly generated as shown in Table 10. Trailer number has been restricted to 9 due to the computational complexity described in Section 3.4. S^2 values for each data set have been calculated by both the optimal model and the DAH as displayed in Figure 23.

Table 10 Number of Data Sets Used for Benchmarking the DAH

<table>
<tr><th colspan="2" rowspan="2">9 trailers</th><th colspan="2"># of Docks</th></tr>
<tr><th>3</th><th>4</th></tr>
<tr><td rowspan="2">Utilization</td><td>High (~85%)</td><td>10</td><td>10</td></tr>
<tr><td>Low (~55%)</td><td>10</td><td>10</td></tr>
</table>

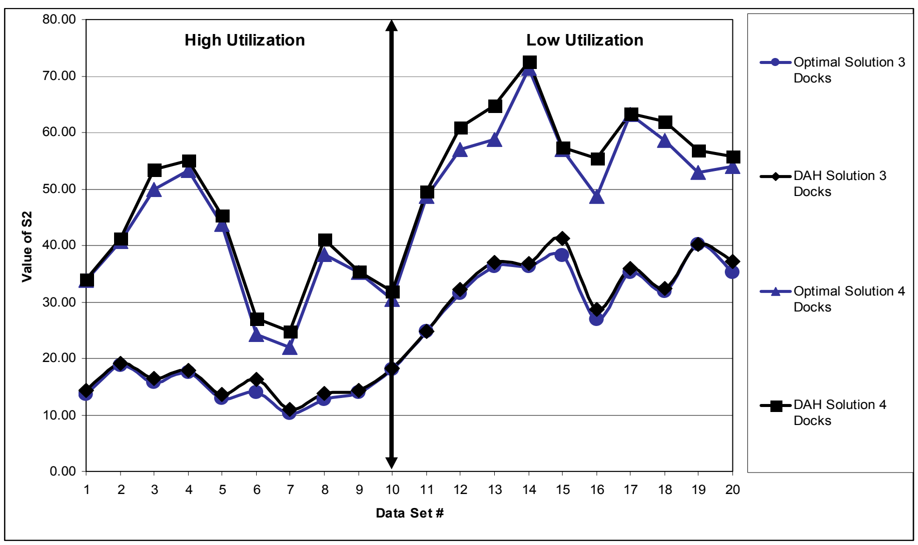

Figure 23 Values Sum of S^2 Values Obtained From Each Data Set

The dock utilizations have been varied by changing the number of docks and changing the service time of trailers. For example, the 3 dock high utilization case has been converted into 4 dock low utilization case with the increasing of dock number by one. The transition from 3 dock high utilization to 3 dock low utilization has been accomplished by decreasing the service times.

Figure 23 also shows that low utilization cases have higher sum of S^2 values than high utilization cases and 4 dock cases have higher sum of S^2 values than 3 dock cases. This is due to the fact that the sum of S^2 values is inherently higher with docks that have higher total idle time.

The performance measure is chosen as *percent deviation* from optimal solution. The percent deviations of sum of S^2 values of the DAH from the sum of S^2 values of the optimal solution are displayed in Figure 24. The DAH on the average performs within 4.41% of the optimal model.

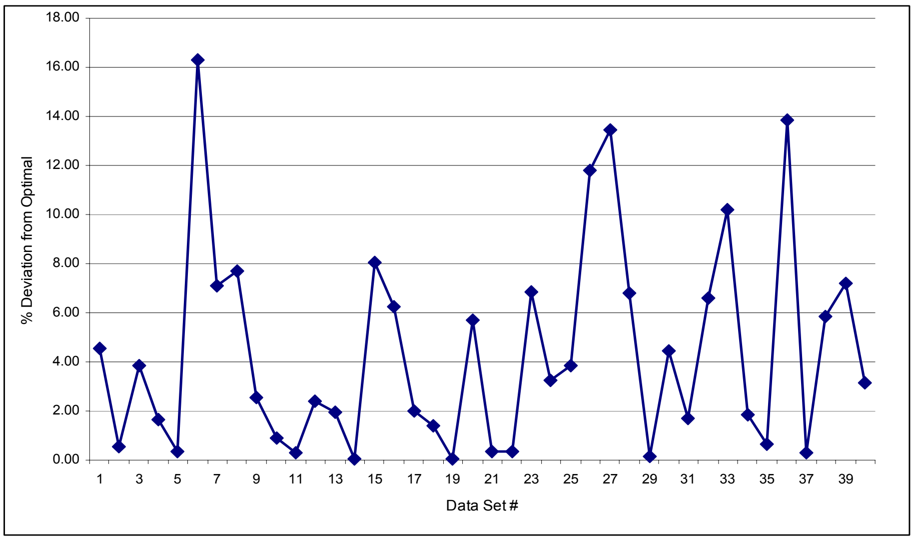

Figure 24 Percent Deviations of Each Data Set from the Optimal Values

4.3 Effects of # of Docks and Dock Utilization on the Performance of the DAH

The objective of this section is to investigate general applicability of the DAH. The internal characteristics of the DAH such as sequential assignments bring the need to investigate this area.

4.3.1 Effect of Number of Docks

Tukey's test is used to test whether there is a significant difference in the mean percent deviation for different number of available docks. Table 11 shows the test data, representing 20 data sets for dock size of two, three and four.

Table 11 Percent Deviations from Optimal Solution for 20 Data Sets

Sets	1	2	3	4	5	6	7	8	9	10
2 Docks	4.52	10.8	1.11	0.6	4.07	0.79	10	14.2	2.44	12.6
3 Docks	4.53	0.57	3.85	1.66	1	16.3	7.09	7.69	2.57	0.91
4 Docks	3.5	6.59	10.2	1.85	0.67	13.8	0.31	5.84	7.19	3.16
Sets(Contd.)	11	12	13	14	15	16	17	18	19	20
2 Docks	9.86	2.47	5.32	3.83	1.23	6.36	0.21	6.65	0.57	2.07
3 Docks	0.3	2.39	1.93	0.05	8.03	6.26	1.99	1.39	0.03	5.72
4 Docks	0.36	1	6.83	3.26	6.59	11.8	13.5	6.79	0.16	4.47

Table 12 ANOVA for Comparison of Means

Analysis of Variance					
Source of Variability	DF	SS	MS	F	P-Value
Dock	2	30.7	15.3	0.86	0.429
Error	57	1016.8	17.8		
Total	59	1047.4			

Table 12 displays the SS (sum of squares) and MS (mean square) values for the three levels and the error. The value of the studentized range statistic q at a significance level of α = 0.05 and (3, 57) degrees of freedom (3 and 57 correspond to the number of levels and the number of degrees of freedom associated with the MS_E respectively):

$$q_{0.05}(3,57) = 3.40$$

The corresponding $T_{0.05}$ value is found as:

$$T_{0.05} = q_{0.05}\sqrt{\frac{MS_E}{n}} = 8.28$$

The differences in averages are:

$$\bar{y}_{1.} - \bar{y}_{2.} = 1.27$$
$$\bar{y}_{1.} - \bar{y}_{3.} = -0.41$$
$$\bar{y}_{2.} - \bar{y}_{3.} = -1.68$$

Since none of the differences is greater in absolute value than the $T_{0.05}$ value, it is concluded that increase in the number of docks does not significantly affect the performance of the heuristic for 2, 3 and 4 docks. For larger dock numbers however, the increase in the number of docks was expected to significantly decrease heuristic performance due to the growth of state space as the number of docks grows since the optimal solution has a larger state space to choose from where as the heuristic is restricted to a smaller state space. Further tests with high dock numbers was not possible to perform due to the computational restrictions of the optimal model.

4.3.2 Effect of Utilization Level

The pairwise comparisons were subjected to paired-sample *t*-tests on the difference between means, μ_d . The test on utilization levels considers the hypothesis:

$$H_0 : \mu_d = 0$$
$$H_A : \mu_d \neq 0$$

where $\mu_d = \mu_H - \mu_L$, H and L denote High utilization and Low utilization treatments respectively.

The null hypothesis is accepted at 99% and higher confidence level as shown in Table 13 indicating the effect of utilization level is not statistically significant on the performance of the heuristic. The heuristic performance, on the average, decreased by 0.94% from low to high utilization level.

Table 13 Effect of Changing Utilization of Docks on Heuristic Performance

Utilization	Average % Deviation	% Change	t-stat	Deg.of freedom	CL	t-table value	C.I.
High	4.91	0.94	0.78	19	99%	2.861	insig.
Low	3.97						

4.4 Conclusion

The results show that, on the average, the DAH performs within 4.41% of the optimal solution at the objective of evenly distributing the idle time at docks and is generally applicable. Most important benefit of the DAH was that it was enable to find feasible solutions to moderate size problems. This is an important aspect since the next chapter investigates how robust the objective of evenly distributing idle times at gates is when the system is subject to variability.

CHAPTER 5
TESTING *DAH* FOR ROBUSTNESS

This chapter investigates how the Dock Assignment Heuristic (DAH) manages the variability in system parameters. Specifically, DAH is tested using variability in arrival times, variability in distribution of freight flow in the LTL truck and variability in transfer time.

5.1 Performance Metric for Terminals

The performance measure used is the percent cut-off (PCO) metric that was introduced in Section 1.2.2.2. This metric focuses on service level of the terminal as a whole. It is important that a package that arrives at the terminal makes its outbound dock in a pre-set time, where it will then leave the terminal. During the time that a package is at the terminal there are many factors that affect the transition of the package directly or indirectly. PCO is a cumulative assessment of these factors.

The PCO metric works in the following way. First, a cut-off time is set for truck arrivals. This is referred to as the arrival cut-off time (ACOT). Another cut-off time is set for the completion of service for all trailers that arrived before ACOT which is referred to as the final cut-off time (FCOT). Any truck from the beginning of the planning horizon to FCOT is planned to be processed in this duration. Freight that passes a set point before FCOT increases the PCO. The ACOT and FCOT differ according to unique terminal characteristics and service level goals.

5.2 Simulation Framework

A simulation model with 50 trailers and 10 docks (5 inbound and 5 outbound) is built (shown in Figure 25 and Figure 26) using Arena Simulation v7.01 by Rockwell Software [88].

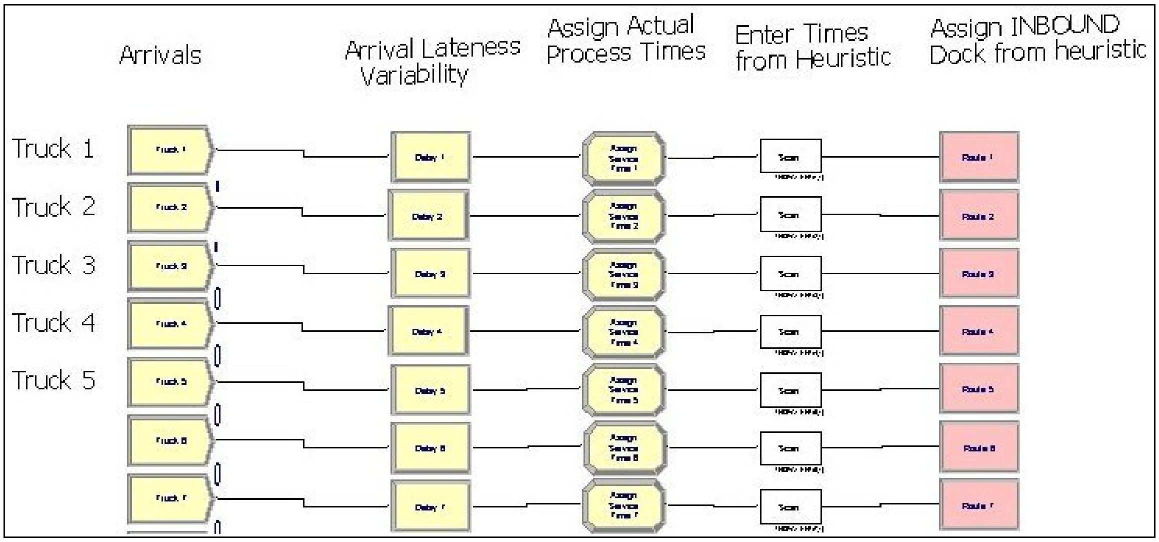

Figure 25 Arrival Process of the Simulation Model Showing Several Trucks

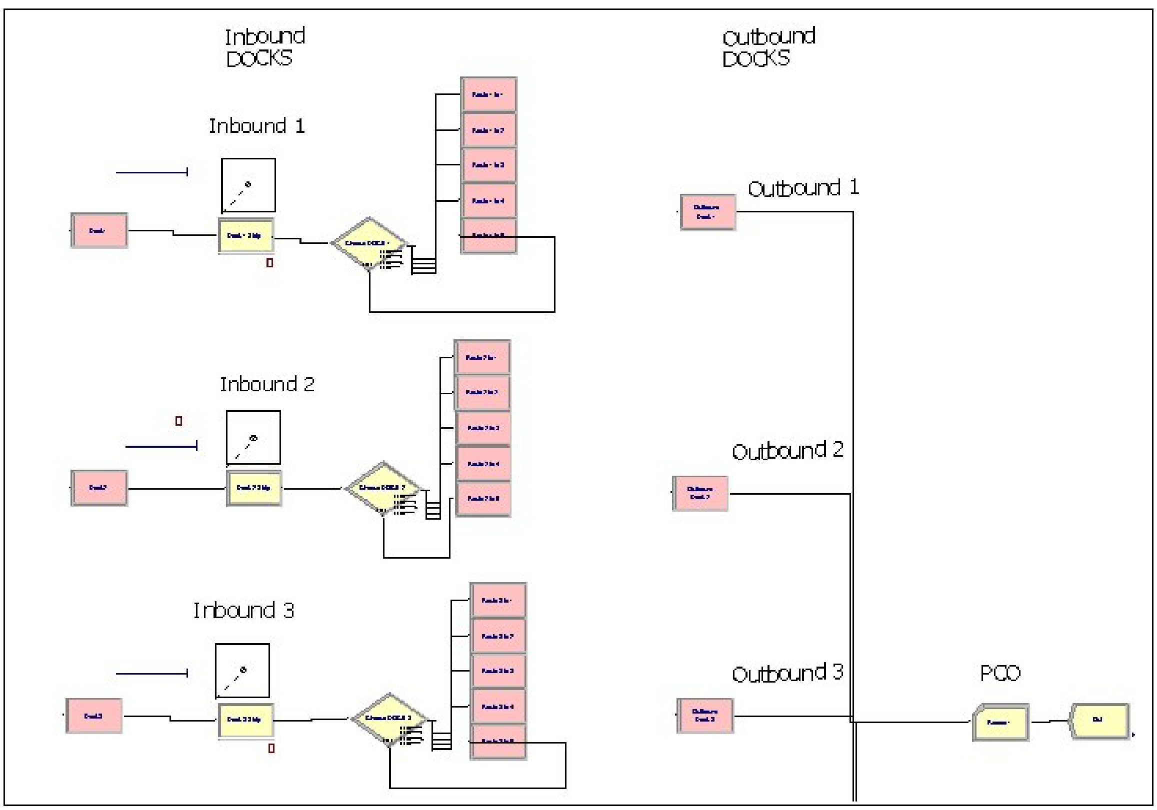

Figure 26 Decision Process of Simulation Model Showing 3 Docks

5.2.1 Simulation Input Data

Data input to simulation model comprise scheduled arrival times and service times. The model uses enter times $E_{j,k}$ and dock assignment information obtained from the solution of the DAH. Data obtained from Watkins Motor Lines [89] for arrival distribution is shown in Table 14. 10 sets of 50 trailer data sets are generated. Figure 27 displays the probability density function generated using the data from Watkins Motor Lines [89].

Table 14 Arrival Data Probability Distribution

hrs			hrs		
from	to	prob.	from	to	prob.
17	18	0.0427	5	6	0.0568
18	19	0.0392	6	7	0.0538
19	20	0.0339	7	8	0.0515
20	21	0.0386	8	9	0.0386
21	22	0.0421	9	10	0.0205
22	23	0.0439	10	11	0.0322
23	24	0.0521	11	12	0.0217
24	1	0.0527	12	13	0.0234
1	2	0.0614	13	14	0
2	3	0.0761	14	15	0
3	4	0.0644	15	16	0
4	5	0.0708	16	17	0

ACOT has been set to 20^{th} unit time and FCOT to 24^{th} unit time. It is planned that all the trucks that arrive between time 0^{th} and 20^{th} unit time will be processed before 24^{th} unit time. Any truck that comes between ACOT and FCOT will be assigned to the next day's PCO. In the simulation these trucks are considered to have $Aj = 0$.

The ACOT and FCOT correspond to 13:00 hrs and 17:00 hrs of the day that supplied the data. This means that PCO will be increased by any freight that arrives between 13:00 the previous day and 13:00 today and make it to its outbound dock before 17:00 today.

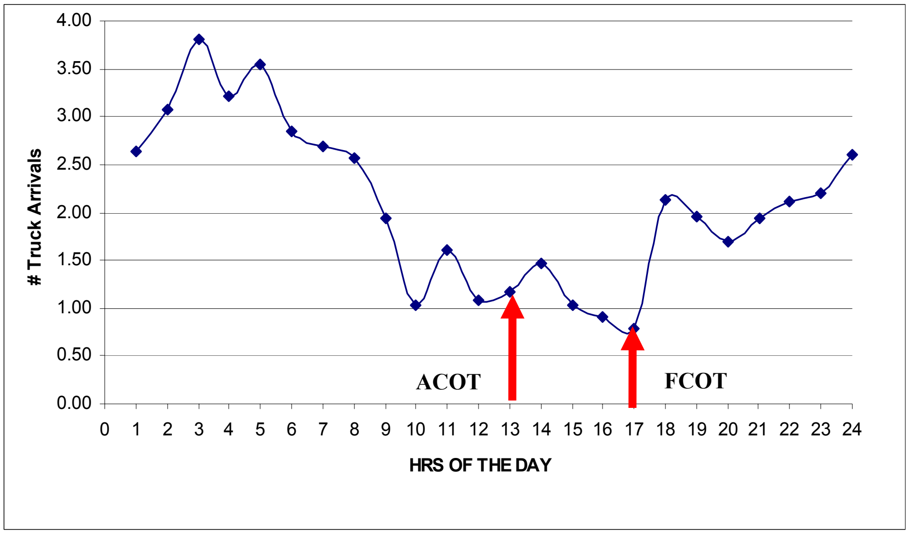

Figure 27 Arrival Probability Distribution

Service time in the terminal refers to two types of operations:

1. Unloading of trailers: Time needed to completely unload the contents of a trailer. Analysis of data supplied from the industry shows that this is exponentially distributed with a mean of 2.11 hours as depicted in Table 15.
2. Transfer of freight from the inbound to the outbound dock: The time it takes for the pallets to arrive at their destination dock.

Table 15 Data Used in Simulation for Service Times

`	Distribution	Parameter
Unloading Time(hrs)	Exponential	2.11
Transfer Time(hrs)	Constant	Low 0.10,0.11,0.12,0,13,0.14
		High 0.20,0.22,0.24,0,26,0.28

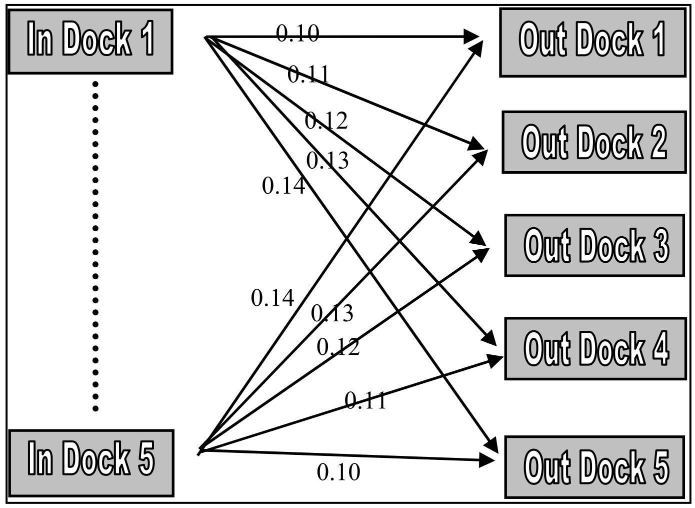

Figure 28 Transfer Times Assigned for Low Setting

The transfer times are input to the simulation model as constants. The two treatments of transfer times are high and low.

5.3 Experimental Set-up

The experimental set-up consists of 20 data sets and 3 treatments:

1. Variability in arrival pattern (*Arrival Lateness SD*): Low standard deviation means less variability in incoming truck arrival times with respect to scheduled arrival times. (SD $\approx$ 0.5) whereas high standard deviation means high variability. (SD $\approx$ 3)
2. Inbound freight destination distribution (*Load Factor*): When the destinations associated with the incoming freight is balanced, the freight in a truck is equally distributed among the outbound trucks that it is transferred to. When it is unbalanced, the flow among the freight to one or more outbound docks is increased, resulting in a shift of load to certain doors.
3. Transfer Time (*Transfer Time*): The time it takes to transfer freight to outbound docks. Transfer time factor is incorporated in the setup to represent the differences between the transferring capabilities of terminals. High setting indicates that the transfer times take long time, increasing the effect of the transfer time factor, and low setting indicates low transfer times, such as but not restricted to, smaller terminals.

5.4 Experimental Results

The following sections discuss the experimental results with respect to afore mentioned three factors.

5.4.1 Variability in Arrival Patterns

Trucks can be subject to significant variability in their arrival times. The DAH is subjected to changes in the standard deviation in the truck arrival distribution. It is assumed that trucks arrive with lateness normally distributed with a mean of zero. By changing the standard deviation of the underlying distribution, the degree of variability in truck arrival times is increased. The results are summarized in Table 16.

Table 16 PCO of DAH with Changing SD in Arrival Lateness

Utilization %	88	80	75	62	76.25
SD	Set 1	Set 2	Set 3	Set 4	Average
0.5	93.66	97.80	100.00	100.00	97.87
1	93.62	97.76	100.00	99.98	97.84
1.5	93.62	97.44	99.94	99.98	97.75
2	92.96	96.42	99.24	99.92	97.14
2.5	91.86	95.22	98.32	99.72	96.28
3	90.62	93.82	96.84	99.36	95.16
3.5	89.06	92.48	94.98	99.00	93.88
4	88.12	90.10	93.36	98.66	92.56
4.5	86.38	87.88	91.60	98.18	91.01
5	85.06	85.48	90.06	97.32	89.48

The graph in Figure 29 shows that the DAH is robust against the variability in arrival times. An average decrease of 8.4% in PCO is observed for. Data obtained from industry show that SD of arrival lateness is approximately 1.5, a level where the quality of the solution obtained using the DAH is virtually unaffected. SD of arrival lateness can vary according to the different characteristics of the terminal, road conditions and seasons, however the DAH proves robust even under significant SD's.

Another observation that can be made from Figure 29 is that lower utilization levels compensate variability better, an expected outcome due to the characteristics of PCO metric which is based on a pre-determined FCOT.

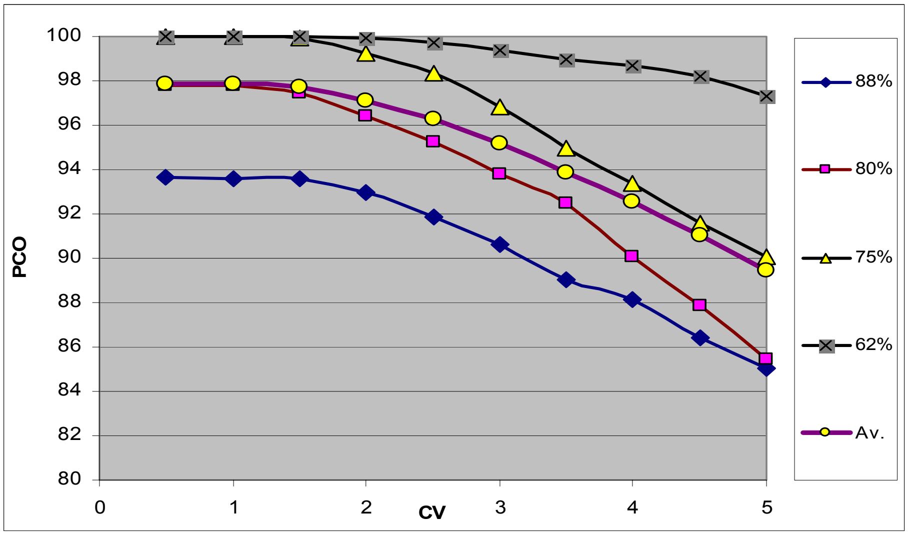

Figure 29 The DAH Subject to Different SD's of Arrival Lateness

5.4.2 Variability in Load Distribution

As mentioned in Chapter 4, the DAH is independent of flow information, thus results are robust against changes in load distribution. The "load factor" had been set to distributions given in Table 17 , where a factor of zero indicates that the load is distributed at docks in balance and a factor of 5 indicates there is significant shift of demand to certain doors.

Table 17 Load Factors and Their Effect on PCO Metric of 4 Arbitrary Data Sets

	Load Distribution					PCO			
Factor	Dock 1	Dock 2	Dock 3	Dock 4	Dock 5	Set 1	Set 2	Set 3	Set 4
0	0.2	0.2	0.2	0.2	0.2	93.92	98.00	92.00	100.00
1	0.35	0.25	0.2	0.15	0.05	93.92	98.00	92.00	100.00
2	0.45	0.3	0.15	0.07	0.03	93.92	98.00	92.00	100.00
3	0.55	0.35	0.05	0.03	0.02	93.92	98.00	92.00	100.00
4	0.65	0.2	0.05	0.05	0.05	93.92	98.00	92.00	100.00
5	0.75	0.15	0.05	0.04	0.01	93.92	98.00	92.00	100.00

5.4.3 Variability in Transfer Time

The DHA's independence of distance and flow vectors may be a disadvantage against dock assignment algorithms utilizing distance minimization formulas, therefore the DHA is subjected to variability in transfer times.

Table **18** displays the times it takes between inbound and outbound docks in hours. The times in

Table **18** have a time factor of 0. Time factor 5 indicates the multiplication of these values by 5.

Table 18 Times between Docks in Hours with Time Factor 0

		Outbound Docks				
		1	2	3	4	5
Inbound Docks	1	0.05	0.06	0.07	0.08	0.09
	2	0.06	0.05	0.06	0.07	0.08
	3	0.07	0.06	0.05	0.06	0.07
	4	0.08	0.07	0.06	0.05	0.06
	5	0.09	0.08	0.07	0.06	0.05

DHA yields approximately a 4% decrease in performance. For especially small transfer times terminals (time factors 0 and 1) the DAH is unaffected as displayed in Figure 30.

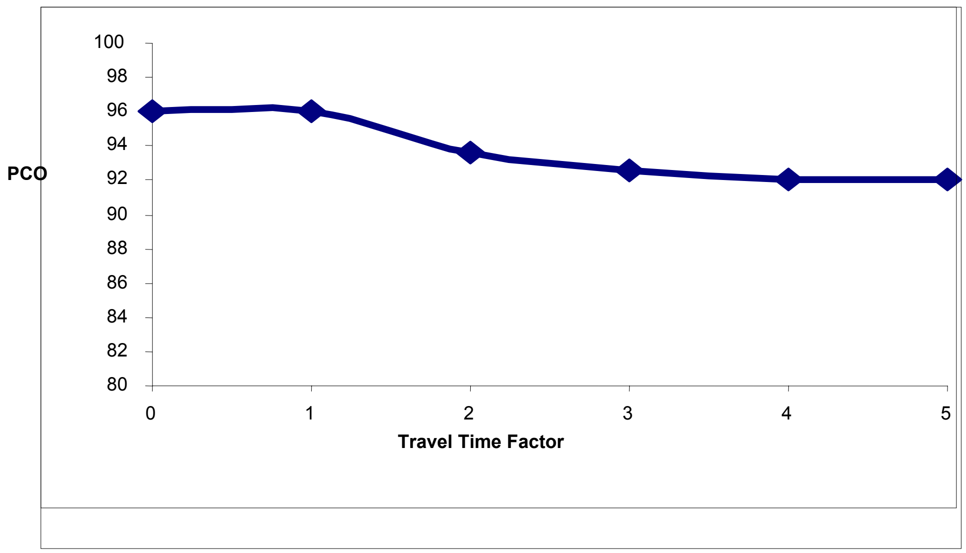

Figure 30 Effect of Travel Time on PCO

5.5 Conclusion

In this chapter the DAH has been tested under varying system parameters, variability in lateness of arrivals, load distribution factors and transfer times. Under these test conditions the DAH performs robust considering PCO metric as the measure of performance.

Although the DAH made it possible to achieve solutions for practical size problems, it continues to be a planning level heuristic. The next chapter introduces a dynamic hybrid heuristic that can be implemented in real time.

CHAPTER 6
A MODEL FOR REAL TIME ASSIGNMENTS

6.1 Introduction

The optimal model introduced in Chapter 3, the distance minimization model of Tsui and Chang described in Section 2.6 and many existing models in literature are planning level models. Suboptimal solutions such as the DAH are computationally tractable; however, they are not real time decision processes.

This chapter introduces a new heuristic that incorporates the benefits the distance minimization models and robustness of DAH. A modification to the DAH and incorporating a decision process for distance minimization yield a hybrid heuristic that can be implemented in real time. The heuristic will be referred to as the *dynamic dock assignment heuristic* (DDAH). Later in the chapter DDAH is compared with DAH for robustness under varying system conditions.

6.2 The Dynamic Dock Assignment Heuristic (DDAH)

As mentioned in the previous chapters, the assignment steps (steps 1, 2 and 3) of DAH results in evenly distributed idle times at each dock. At the operational level, this means that the utilizations of existing docks are similar which is valuable for robustness. DDAH also incorporates the distance minimization objective by considering the flow and distance information yielding a hybrid heuristic that aims to combine the advantages of both approaches. The steps of the DDAH are as follows:

Step 1: Select the trailer with the earliest actual arrival time.
Step 2: Based on the "freight distribution" select the idle inbound dock that minimizes travel time to outbound dock. If no docks are available, wait until a dock becomes available.
Step 3: Repeat steps 1 and 2 for all available trailers.

Step 1 of DAH sorts the trailers in ascending order of Estimated Arrival Times. At operational level, this is analogous to building a single queue for first come first serve processing where the trailers will be selected according to their arrival order. DDAH selects the trailer with the earliest actual arrival time, or the first trailer in the queue. Step 2 of DAH assigns all the trailers sequentially to the dock with the smallest existing leave time. At operational level this is equivalent to assigning the trailer to the first available dock (waiting until a dock becomes available) since the first available dock is the dock with the smallest leave time. Step 3 of DAH is updating the enter times once the trailer is assigned to a dock. At operational level, this is analogous to assigning the actual enter time.

6.3 DDAH vs. DAH

This section compares the performance of the DDAH to DAH following the experimental set up shown in Figure 31. Similar to the experimental design used for testing DAH in Chapter 5, the DDAH is subjected to variability in system conditions and simultaneously compared to the DAH using the PCO metric. The decision process of the simulation model is modified to implement the DDAH as shown in Figure 32.

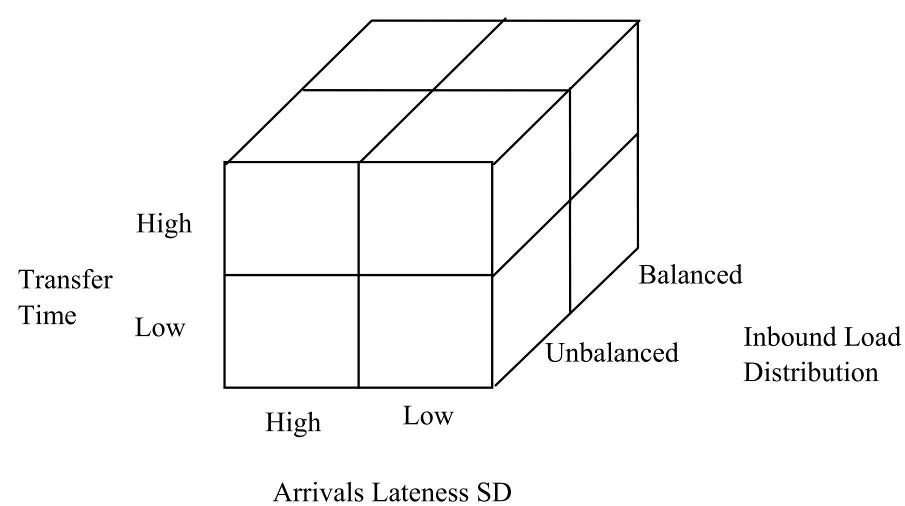

Figure 31 Variables Used in Experimental Design

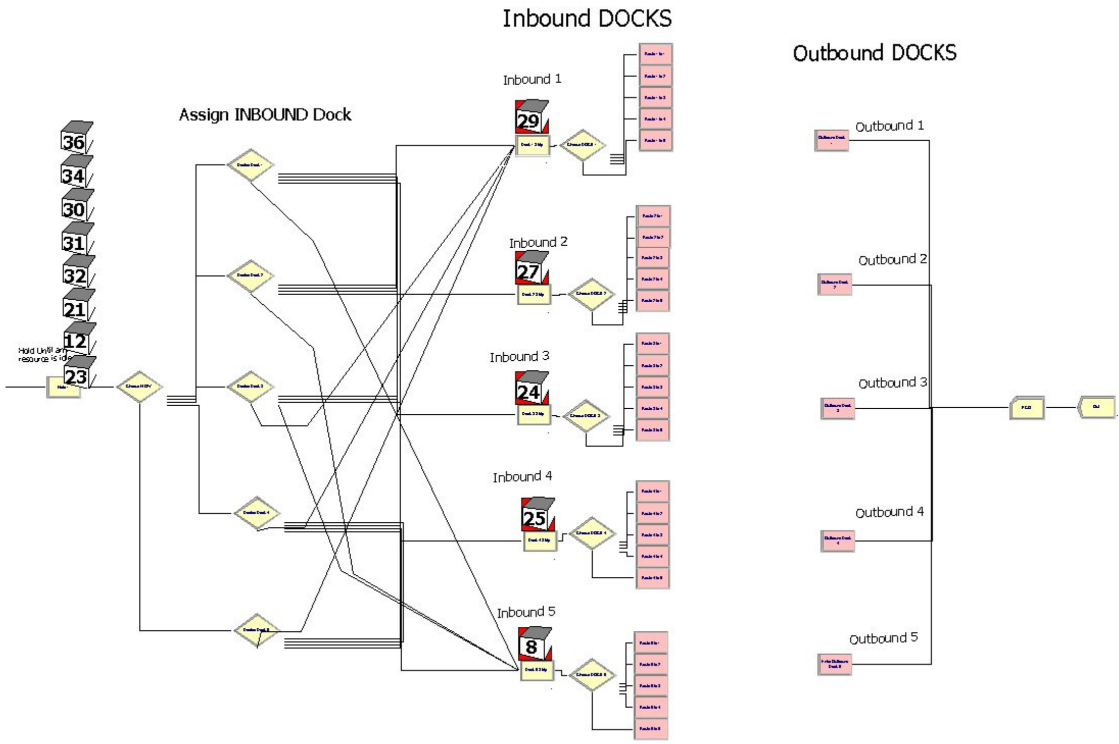

Figure 32 Screenshot of the Decision Process of the DDAH Simulation Model

The trailer is initially assigned to the inbound dock that minimizes the total travel distance of the trailer contents to the outbound dock. For example, assume a trailer is initially assigned to inbound dock 1 since the total travel distance of its contents is minimized at the inbound dock closest to outbound dock 1. If outbound dock 1 is busy, then the trailer is first re-assigned to inbound dock 2, and then inbound dock 3 and so on. For the simulation it is assumed that the load distributions in each truck are known and the initial assignments are made based on the demand for outbound docks.

Table 19 Levels of Factors Used in the Experiment

	Transfer Time	Arr. Lateness SD	Load Distribution
Low	Factor 1	0.5	0.2,0.2,0.2,0.2,0.2
High	Factor 3	3	0.40,0.25,0.15,0.10,0.10

During the experimentation the DDAH outperformed the DAH by a small margin based on the PCO metric as shown in Figure 33. This was anticipated since DDAH does not delay the assignment of incoming trailers as is the case with DDAH, however the purpose of testing for robustness is to investigate how much the performance of the heuristic deviates from its performance when the system is not subject to variability. The range of the averages displayed in Table 20 show that DDAH are not significantly affected. The range for the DDAH is 1.26 versus 2.36 for the DAH for the test cases considered.

Table 20 Table Showing the Effects of Treatments on PCO Metric

Treatment				PCO	
Run #	Transfer Time	Arrival SD	Load Dist.	DDAH	DAH
1	Low	Low	Low	98.40	97.62
2	Low	Low	High	97.95	96.78
3	Low	High	Low	97.88	97.10
4	Low	High	High	97.96	96.34
5	High	Low	Low	97.72	95.99
6	High	Low	High	97.14	95.92
7	High	High	Low	97.42	95.78
8	High	High	High	97.23	95.26
			Range	1.26	2.36

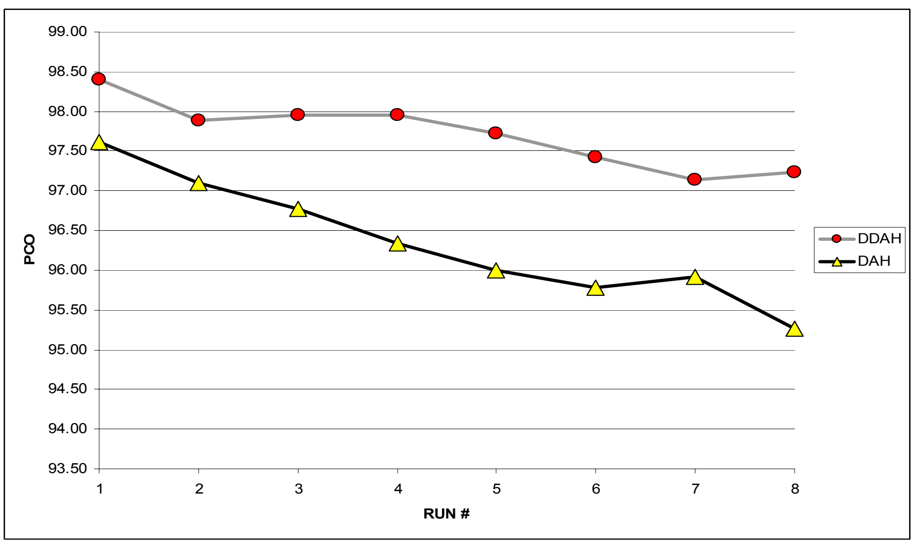

Figure 33 Changes in PCO with Different Treatments

The results indicate that DDAH is a robust heuristic against system variability when PCO is considered as performance metric. The next chapter compares DDAH with a static dock assignment approach used in the industry using data obtained from Watkins Motor Lines.

CHAPTER 7
CASE STUDY

In this chapter, the DDAH is compared to a dock assignment approach that is used by Watkins Motor Lines which will be referred to as the *static dock assignment model* (SDAM). The dock assignments in SDAM are made based on minimization of travel distance, similar to the model of Tsui and Chang [14] that has been previously presented. In SDAM inbound and outbound docks are assigned based on aggregate data representing several months or longer. The two models are tested under the experimental setup used in Chapter 6.

7.1 Input Data

The terminal under consideration has a total of 49 docks of which 46 are used. 23 docks are allocated to inbound and 23 are allocated to outbound operations as shown in Figure 34. (City names with * denote strip docks)

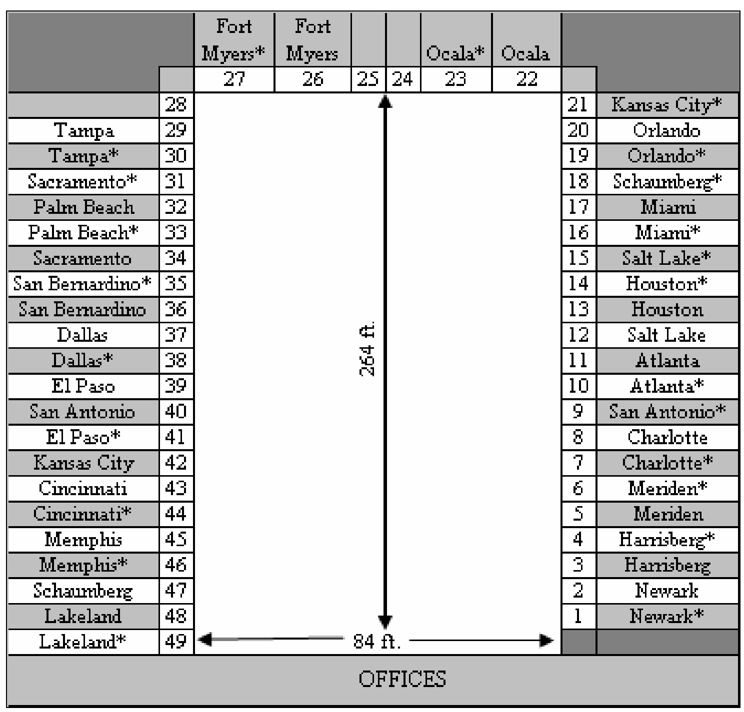

Figure 34 Layout of the Terminal Used for the Case Study [90]

The terminal is 264 ft. in length and 84 ft. in width with 12 ft. of space between adjacent docks. Using the distance information travel times between doors have been calculated as shown in Figure 35 with the city codes given in Table 21.

Table 21 City Codes and Corresponding Dock Numbers

Dock	1	2	3	4	5	6	7	8	9	10	11	12
City	NWK	HBG	MCT	CHA	ATL	SLC	HOU	MIA	ORL	OCA	FTM	TPA
Dock	13	14	15	16	17	18	19	20	21	22	23	24
City	PBC	SAC	SBR	DLS	ELP	SAT	KCY	CIN	MPS	SCH	LAK	

INBOUND \ OUTBOUND	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	0.24	0.48	0.96	1.68	2.40	2.64	2.88	3.84	4.56	5.40	6.12	6.24	5.52	5.04	4.56	4.32	3.84	3.60	3.12	2.88	2.40	1.92	1.68
2	0.48	0.24	0.24	0.96	1.68	1.92	2.16	3.12	3.84	4.92	5.64	5.76	5.04	4.56	4.08	3.84	3.36	3.12	2.64	2.40	1.92	1.92	2.16
3	0.96	0.72	0.24	0.48	1.20	1.44	1.68	2.64	3.36	4.20	4.92	5.04	4.32	3.84	3.36	3.12	2.64	2.40	1.92	1.68	2.16	2.64	2.88
4	1.20	0.96	0.48	0.24	0.96	1.20	1.44	2.40	3.12	3.96	4.68	4.80	4.08	3.60	3.12	2.88	2.40	2.16	1.68	1.92	2.40	2.88	3.12
5	1.92	1.68	1.20	0.48	0.24	0.48	0.72	1.68	2.40	3.24	3.96	4.08	3.36	2.88	2.40	2.16	1.68	1.92	2.40	2.64	3.12	3.60	3.84
6	3.12	2.88	2.40	1.68	0.96	0.72	0.48	0.48	1.20	2.04	2.76	2.88	2.16	1.68	2.16	2.40	2.88	3.12	3.60	3.84	4.32	4.80	5.04
7	2.88	2.64	2.16	1.44	0.72	0.48	0.24	0.72	1.44	2.28	3.00	3.12	2.40	1.92	1.92	2.16	2.64	2.88	3.36	3.60	4.08	4.56	4.80
8	3.36	3.12	2.64	1.92	1.20	0.96	0.72	0.24	0.96	1.84	2.52	2.64	1.92	1.92	2.40	2.64	3.12	3.36	3.84	4.08	4.56	5.04	5.28
9	4.08	3.84	3.36	2.64	1.92	1.68	1.44	0.48	0.24	1.20	1.92	1.92	2.16	2.64	3.12	3.36	3.84	4.08	4.56	4.80	5.28	5.76	6.00
10	4.92	4.64	4.20	3.48	2.76	2.52	2.28	1.32	0.60	0.24	0.96	1.80	2.52	3.00	3.48	3.72	4.20	4.44	4.92	5.16	5.64	6.12	6.36
11	6.00	5.76	5.28	4.56	3.84	3.60	3.36	2.40	1.68	0.96	0.24	0.48	1.20	1.68	2.16	2.40	2.88	3.12	3.60	3.84	4.32	4.80	5.04
12	5.76	5.52	5.04	4.32	3.60	3.36	3.12	2.16	1.92	1.80	1.08	0.24	0.48	0.96	1.44	1.68	2.16	2.40	2.88	3.12	3.60	4.08	4.32
13	5.04	4.80	4.32	3.60	2.88	2.64	2.40	1.92	2.64	2.52	1.80	0.96	0.24	0.24	0.72	0.96	1.44	1.68	2.16	2.40	2.88	3.36	3.60
14	5.52	5.28	4.80	4.08	3.36	3.12	2.88	1.92	2.16	2.04	1.32	0.48	0.24	0.72	1.20	1.44	1.92	2.16	2.64	2.88	3.36	3.84	4.08
15	4.56	4.32	3.84	3.12	2.40	2.16	1.92	2.40	3.12	3.00	2.28	1.44	0.72	0.24	0.24	0.48	0.96	1.20	1.68	1.92	2.40	2.88	3.12
16	3.84	3.60	3.12	2.40	1.68	1.92	2.16	3.12	3.84	3.72	3.00	2.16	1.44	0.96	0.48	0.24	0.24	0.48	0.96	1.20	1.68	2.16	2.40
17	3.12	2.88	2.40	1.68	2.40	2.64	2.88	3.84	4.56	4.44	3.72	2.88	2.16	1.68	1.20	0.96	0.48	0.24	0.24	0.48	0.96	1.44	1.68
18	1.68	1.44	0.96	0.24	0.72	0.48	0.96	1.92	2.64	3.48	4.20	4.32	3.60	3.36	2.88	2.40	1.92	1.68	2.16	2.40	2.88	3.36	3.60
19	4.56	4.32	3.84	3.12	2.40	2.16	1.92	0.96	0.24	0.48	1.20	1.68	2.40	2.88	3.36	3.60	4.08	4.32	4.80	5.04	5.52	6.00	6.24
20	2.40	2.16	1.68	2.40	3.12	3.36	3.60	4.56	5.28	5.16	4.44	3.60	2.88	2.40	1.92	1.68	1.20	0.96	0.48	0.24	0.24	0.72	0.96
21	1.92	1.68	2.16	2.88	3.60	3.84	4.08	5.04	5.76	5.64	4.92	4.08	3.36	2.88	2.40	2.16	1.68	1.44	0.96	0.72	0.24	0.24	0.48
22	3.84	3.60	3.12	2.40	1.68	1.44	1.20	0.24	0.48	1.32	2.04	2.16	1.92	2.40	2.88	3.12	3.60	3.84	4.32	4.56	5.04	5.52	5.76
23	2.16	2.40	2.88	3.60	4.32	4.56	4.80	5.76	6.48	6.36	5.64	4.80	4.08	3.60	3.12	2.88	2.40	2.16	1.68	1.44	0.96	0.48	0.24

Figure 35 Transfer Times between Docks in Minutes

The travel times are used to represent transfer time of pallets from the strip door to its stack door. There are two levels of transfer time, low and high. The travel times given in Figure 35 represent low setting and at high setting travel times are twice as much. Both DDAH and SDAM utilize distance minimization objectives, thus it is anticipated that the models will not be affected by the change in transfer time.

The analysis of data obtained from the terminal suggests that truck arrivals have exponential interarrival time with a mean of 12 minutes and the service rate is exponentially distributed with a mean of 2.11 hours.

Table 22 shows the original distribution of freight on outbound docks. There are two levels of load distribution as shown in Figure 36. The original distribution obtained from the data is accepted as a "balanced" distribution since static dock assignments are made to minimize this freight travel distance. The "shifted" distribution represents an unbalanced load distribution.

For the case study the load distribution data for inbound trucks is not available. Therefore, the initial assignments are made based on the *dominant load* in a truck. The dominant loads are determined according to the outbound load distribution. The inbound trucks are assigned a dominant destination load according to the aggregate outbound load data, and later initially assigned to the inbound door that minimizes the distance to the outbound dock of dominant load.

Table 22 Original Freight Distributions for Outbound Docks

Dock	%	Dock	%
1	1.32	13	10.68
2	3.59	14	2.34
3	3.02	15	3.21
4	3.01	16	3.75
5	3.91	17	2.27
6	2.49	18	1.89
7	1.15	19	2.29
8	12.42	20	2.72
9	11.86	21	2.70
10	5.18	22	4.30
11	4.30	23	0.12
12	11.48		

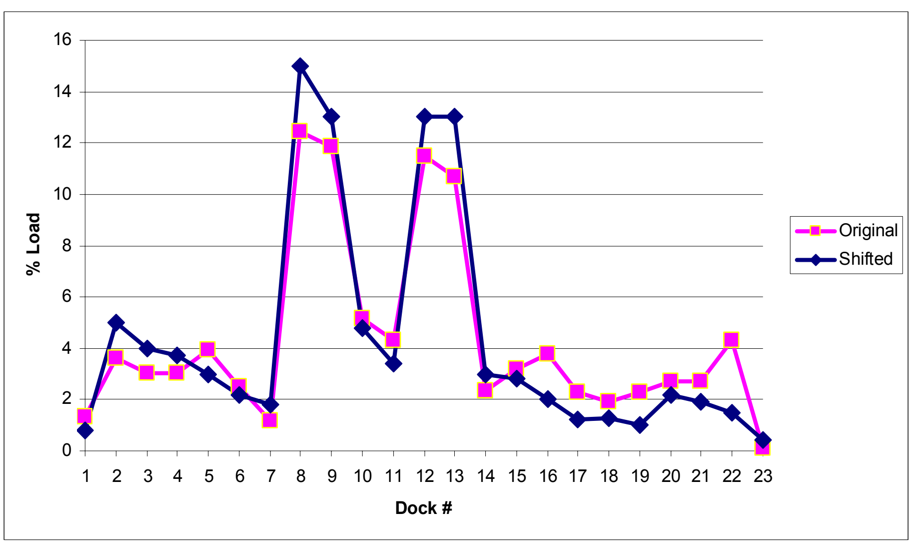

Figure 36 Original and Shifted Freight Load % Distributions

7.2 Simulation Results

The results indicate that DDAH outperforms SDAM based on the PCO metric. The simulation results for low utilization are displayed in Table 23 and for high utilization in Table 24.

The DDAH is less affected then SDAM on the average and individually at each treatment. The deviations in the PCO metric for low and high utilization levels are shown in Table 25 and Table 26, respectively. At low utilization levels the differences in deviations are less significant because at low utilization levels the system is capable of creating enough time to compensate for the losses as depicted in Table 25. System utilization is increased by increasing the number of arrivals by a factor of two. When utilization level is increased, the performance of the DDAH proved more robust as the differences in deviations increased based on PCO metric as displayed in Table 26.

Table 23 Simulation Results Showing PCO for Low Utilization

			PCO	
Forklift Transfer	Flow	Delay CV	DDAH	SDAM
Low	Bal.	Low	97.65	80.73
Low	Bal.	High	92.38	76.38
Low	UnBal	Low	97.64	75.41
Low	UnBal	High	92.40	71.70
High	Bal.	Low	97.62	80.70
High	Bal.	High	92.36	76.36
High	UnBal	Low	96.86	71.70
High	UnBal	High	92.38	71.70

Table 24 Simulation Results Showing PCO for High Utilization

			PCO	
Forklift Transfer	Flow	Delay CV	DDAH	SDAM
Low	Bal.	Low	88.54	58.28
Low	Bal.	High	85.18	56.41
Low	UnBal	Low	88.52	52.79
Low	UnBal	High	85.18	51.13
High	Bal.	Low	88.44	58.28
High	Bal.	High	85.11	56.41
High	UnBal	Low	88.41	52.79
High	UnBal	High	85.07	51.13

Table 25 Percent Deviations from PCO for Low Utilization

Transfer Time	Load Dist.	Arrival SD	DDAH	SDAM
Low	Bal.	Low	PCO Dev.	
Low	Bal.	High	-5.40	-5.40
Low	UnBal	Low	-0.02	-6.59
Low	UnBal	High	-5.38	-11.18
High	Bal.	Low	-0.03	-0.04
High	Bal.	High	-5.42	-5.42
High	UnBal	Low	-0.81	-11.18
High	UnBal	High	-5.40	-11.18
		Average	-3.21	-7.28
		Range	5.40	11.14

The results also point out that DDAH is virtually unaffected from the variability in distribution of load in trucks as denoted by Flow in Table 25 and Table 26. The "range" indicates the difference between the maximum and minimum value of deviation of PCO from the base treatment for the different treatments. Figure 37 depicts the deviations of the results

Table 26 Percent Deviations from PCO for High Utilization

Transfer Time	Load Dist.	Delay SD	DDAH	SDAM
Low	Bal.	Low	PCO Dev.	
Low	Bal.	High	-3.80	-3.21
Low	UnBal	Low	-0.02	-9.42
Low	UnBal	High	-3.80	-12.28
High	Bal.	Low	-0.11	0.00
High	Bal.	High	-3.87	-3.21
High	UnBal	Low	-0.15	-9.42
High	UnBal	High	-3.92	-12.28
		Average	-2.24	-7.11
		Range	3.89	12.28

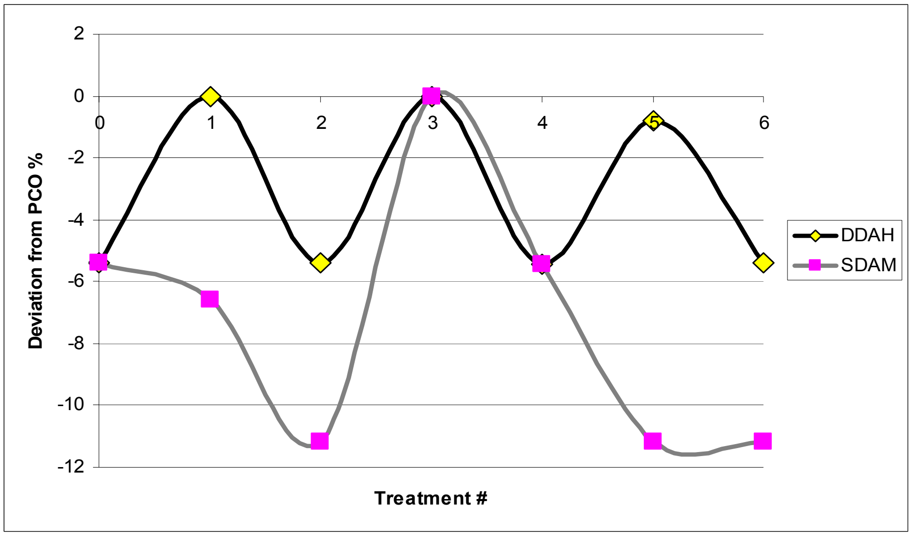

Figure 37 Deviations from PCO as Variability is Introduced to the System for Low Utilization

The effect of transfer time is very insignificant due to the low utilization of the terminal since at low utilizations the system has more capability to compensate the transfer time factor. When the utilization is increased DDAH experiences a slight decrease in its performance. This is due to its inherent characteristic which balances a trade-off between travel distance and robustness.

7.3 Conclusion

A case study using real data from an LTL terminal has been performed to investigate the robustness of the DDAH. The performance of DDAH is compared to the existing static assignment approach of the terminal.

The results show that DDAH is a robust heuristic when compared to the SDAM based on PCO metric, a metric chosen to represent the service levels of the terminal as a whole. DDAH performed equally or better in every treatment introduced to the system. Its performance is also better at higher utilization levels.

CHAPTER 8
CONCLUSIONS AND FUTURE RESEARCH

8.1 Conclusions

This thesis presents a robust dock assignment approach at LTL terminals to minimize the performance decrease when variability is present. Variability is defined in the following three categories:

1. Variability in Truck Arrival Times: Represented by standard deviation of the lateness distribution.
2. Variability in Load Distribution: The distribution of the destination of freight inside the trailer that make a "less-than-truckload" type of load.
3. Variability in Transfer Time: Times that transportation units inside the terminal need to travel. Models utilizing minimization of distance are advantageous in theory in this aspect.

An optimal model is introduced in Chapter 3 as a resource allocation model. The optimal model represents the theory behind the robustness approach, that is, by evenly distributing the total idle time of the docks between the trailers, the system is expected to have buffers to manage variability between the scheduled times and actual times. The computational difficulties associated with the optimal model render realistic size problems unsolvable.

The need for feasible solutions that reflect the robustness approach has lead to the development of a dock assignment heuristic; the DAH. First the DAH has been benchmarked with the existing optimal solutions to investigate its quality. The statistical tests indicated that with small size problems the DAH performs on the average within 4.41% of the optimal model which concluded that the DAH is acceptable for general applicability.

By modifying the DAH a hybrid heuristic that can be implemented in real time is developed, namely the dynamic dock assignment heuristic (DDAH). The DDAH has been tested with the experimental setup used to test the robustness of the DAH. The results show that DDAH is also a robust heuristic, considering the PCO metric.

At the next stage, a design of experiments is performed to test the capabilities of the DAH under system variability. The percent cut-off metric has been adapted as a performance measure, for its representative capabilities of the service level of a terminal as a whole. The results show that the DAH is not significantly affected by the variability in the system, therefore its robustness is accepted. The DAH alleviated computational issues, however not to a level to be implemented in real time, thus a heuristic with real time assignment capabilities is investigated.

The final testing of the DDAH was a case study that used real data supplied by Watkins Motor Lines comparing with the existing static dock assignment model (SDAM) of the terminal. The results agree that DDAH is robust when compared to the SDAM based on PCO metric. DDAH performed equally or better than SDAM in every treatment introduced to the system. It is also shown that it performs well at low and high utilization levels.

8.2 Future Research

Some of the potential areas for future research include:

The reformulation of the optimal model: Increasing the efficiency of the optimal model by reformulating the model such as substituting the Y_{ijk} variable. Optimal solutions to larger size problems will provide better comparing grounds for heuristics.

Artificial Intelligence Approaches: Techniques mentioned in the literature review are potential solution methodologies for significant improvement in computation times with acceptable trade offs in solution quality.

Increasing the Level of Uncertainty: Introducing new sources of variability to the system by examining the terminals in more detail would help increase robustness in practical sense.

The Economics of The Optimal Model and DAH: Since both models are planning level models and they extend the duration of service times, their impact on the economy of the terminal hold potential for savings for labor.

REFERENCES

[1] Nationmaster Statistics. http:// www.nationmaster.com.

[2] U.S. Census Bureau. http://www.census.gov.

[3] T.M. Corsi, C.M. Grimm and J.N. Feitler. Measuring Firm Strategic Change in the Regulated Motor Carrier Industry: An 18 Year Evaluation. *Transportation Research E*, 33 (3):159-169, 1997.

[4] http://tmip.fhwa.dot.gov/clearinghouse/docs/quick/append_a.pdf.

[5] J.W. Braklow, W.W. Graham, S.M. Hassler, K.E. Peck and W.B. Powell. Interactive Optimization Improves Service and Performance for Yellow Freight System. *Interfaces*, 22(1): 147-172, 1992.

[6] A. Kanafani and A.A. Ghobrial. Airline Hubbing--Some Implications for Airport Economics. *Transportation Research*, 19 (A): 15-27, 1985.

[7] J.F.Campbell, Integer Programming Formulations of Discrete Hub Location Problems. *European Journal of Operational Research*, 72: 387-405, 1994.

[8] R. K. Martin. Large Scale Linear and Integer Optimization: A Unified Approach. *Kluwer Academic Publishers*. 1999.

[9] C. Barnhart, A. Farahat, and M. Lohatepanont. Airline Fleet Assignment: An Alternative Model and Solution Approach Based on Composite Variables. *Operations Research (forthcoming),* 2001.

[10] A. Armacost, C. Barnhart and K. Ware. Composite Variable Formulations for Express Shipment Service Network Design. *Transportation Science*, 36(1): 1-20, 2002.

[11] K.R. Gue and J.J. Bartholdi. Reducing Labor Costs in an LTL Crossdocking Terminal. *Operations Research*, 48 (6): 823-832, 2000.

[12] K.R. Gue. The Effects of Trailer Scheduling on the Layout of Freight Terminals. *Transportation Science*, 33 (4): 419-428, 1999.

[13] J.J. Bartholdi and K.R. Gue. The Best Shape for a Crossdock, 2003.

[14] L. Y. Tsui and C.H. Chang. Optimal Solution to a Dock Door Assignment Problem. *Computers & Industrial Engineering*, 23: 283–286, 1992.

[15] K. Gue. A Note on Performance Metrics for Warehouses, 1999.

[16] http://tli.isye.gatech.edu/news/article.php?id=55.

[17] A. Bolat. Procedures for Providing Robust Gate Assignments for Arriving Aircrafts. *European Journal of Operational Research*, 120: 63-80, 2000.

[18] C.R. Reeves. Modern Heuristic Techniques for Combinatorial Problems. *John Wiley & Sons*. 1993.

[19] C.H. Papadimitriou and K.Steglitz. Combinatorial Optimization: Algorithms and Complexity. *Prentice-Hal, Englewood Cliffs, New Jersey,* 1982.

[20] G.L. Nemhauser and L.A. Wolsey. Integer and Combinatorial Optimization. *Wiley Interscience series in discrete mathematics and optimization. John Wiley & Sons, Inc.* 1988.

[21] L.D. Gaspero. Local Search Techniques for Scheduling Problems: Algorithms and Software Tools. *PhD Thesis, University of Udine, Italy,* 2002.

[22] R.E. Burkard and E. Cela, Linear Assignment Problems and Extensions, in P.M. Pardalos and D.Z. Du (Ed.). Handbook of Combinatorial Optimization. *Dordreck: Kluwer Academic Publishers*. 75-149, 1999.

[23] P.M. Pardalos, F. Rendl, and H. Wolkowicz. The Quadratic Assignment Problem: A Survey And Recent Developments*. DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 16: 1-42, 1994.

[24] http://www-unix.mcs.anl.gov/otc/Guide/CaseStudies/qap/.

[25] A.M. Geoffrion and G.W. Graves. Scheduling Parallel Production Lines with Changeover Costs: Practical Applications of Quadratic Assignment / LP Approach. *Operations Research,* 24: 595-610, 1976.

[26] L. Steinberg. The Backboard Wiring Problem: A Placement Algorithm. *SIAM Review,* 3: 37-50, 1961.

[27] Assignment Problems in Parallel And Distributed Computing, Kluwer Academic Publishers, Boston, 1987.

[28] L.J. Hubert. Assignment Methods in Combinatorial Data Analysis. *Marcel Dekker, Inc., New York, NY 10016,* 1987.

[29] R.L. Francis and J.A. White. Facility Layout and Location. *Prentice-Hall Englewood Cliffs, N.J.,* 1974.

[30] J. Krarup and P.M. Pruzan. Computer Added Layout Design. *Mathematical Programming Study,* 9: 75-94, 1978.

[31] E.J. McCormik. Human Factors Engineering. *McGraw-Hill, New York,* 1970.

[32] A. Haghani and C. Min-Cheng. Optimizing Gate Assignments at Airport Terminals, *Transportation Research A*, 32 (6): 437-454, 2001.

[33] T. C. Koopmans and M. J. Berkmann. Assignment Problems and The Location Of Economic Activities. *Econometrica*, 25: 53–76, 1957.

[34] C. Chang. Flight Sequencing And Gate Assignment In Airport Hubs. *Ph.D. Dissertation, University of Maryland at Colege Park,* 1994.

[35] J.P. Braaksma. Reducing Walking Distances at Existing Airports. *Airport Forum,* 4, 1977.

[36] O. Babic, D. Teodorović, V. Tošic. Aircraft Stand Assignment to Minimize Walking. *Journal of Transportation Engineering*, 110: 55-66, 1984.

[37] R.S. Mangoubi, and D.F.X. Mathaisel. Optimizing Gate Assignments at Airport Terminals. Transportation Science, 19(2):173-188, 1985.

[38] R.A. Bihr. A Conceptual Solution to the Aircraft Gate Assignment Problem Using 0,1 Linear Programming. *Computers and Industrial Engineering* 19: 280-284, 1990.

[39] P. Baron. A Simulation Analysis of Airport Terminal Operations. *Transportation Research.* 3 (4): 481-491, 1969.

[40] H. Ding, A. Lim, B. Rodrigues and Y. Zhu. The Over-Constrained Airport Gate Assignment Problem. *Computers and Operations Research* (to appear), 2004.

[41] J. Xu and G. Bailey. The Airport Gate Assignment Problem: Mathematical Model and a Tabu Search Algorithm. *Proceedings of the 34th Annual Hawaii International Conference on System Sciences,* 3 (3), 2001.

[42] A. Lim, B. Rodrigues and Y. Zhu. Aircraft and Gate Assignment Problem with Time Windows. *15th IEEE International Conference on Tools with Artificial Intelligence*, 2003.

[43] G.D. Gosling. Design of An Expert System For Aircraft Gate Assignment. *Transportation Research A,* 24 (1): 59-69, 1990.

[44] K. Srihari and R. Muthukrishnan. An Expert System Methodology For An Aircraft Gate Assignment. *Computers and Industrial Engineering,* 21 (1-4): 101-105, 1991.

[45] M.I. Hassounah and G.N. Steuart. Demand For Aircraft Gates. *Transportation research record.* 1423: 26-33, 1993.

[46] S. Yan, and C. Chang. A Network Model for Gate Assignment. *Journal of Advanced Transportation*, 32(2): 176-189, 1998.

[47] S. Yan, and C. Chang. Optimization of Multiple Objective Gate Assignments. *Transportation Research*, 35A (5): 413-432, 2001.

[48] S. Sahni and T. Gonzalez. P-Complete Approximation Problems. *Journal of the Association of Computing Machinery*, 23: 555–565, 1976.

[49] O. Kettani and M. ORAL. Reformulating Quadratic Assignment Problems for Efficient Optimization, *IIE Transactions*, 25 (6): 97-107, 1993.

[50] QAPLIB. http://www.opt.math.tu-graz.ac.at/qaplib/.

[51] N. Christofides and E.Benavent. An Exact Algorithm for The Quadratic Assignment Problem. *Operations Research,* 37 (5): 760-768, 1989.

[52] M.S. Bazaraa and H.D. Sherali. Bender's Partitioning Scheme Applied to a New Formulation Of The Quadratic Assignment Problem. *Naval research logistics quarterly*, 27: 29-41, 1980.

[53] P.M Pardalos and J.V. Crouse. A Parallel Algorithm For The Quadratic Assignment Problem. *Proceedings of the ACM/IEEE conference on Supercomputing.*1989.

[54] C. Roucairol. A Parallel Branch and Bound Algorithm For The Quadratic Assignment Problem. *Discrete applied mathematics,* 18: 211-225, 1987.

[55] E.L. Lawler. The Quadratic Assignment Problem. *Management Science.* 9 (4): 586-599, 1963.

[56] A.M. Frieze and J. Yadegar. On the Quadratic Assignment Problem. *Discrete applied mathematics.* 5: 89-98, 1983.

[57] T. Mautor and C. Roucairol. A New Exact Algorithm for The Solution Of Quadratic Assignment Problems. *Discrete Applied Mathematics*, 55: 281-293, 1994.

[58] N. Christofides and E. Benavent. An Exact Algorithm for The Quadratic Assignment Problem On A Tree. *Operations Research.* 37 (5): 760-768, 1989.

[59] B. Chen. Special Cases Of The Quadratic Assignment Problem. *European journal of operations research,* 81: 410-419, 1995.

[60] V.G. Deineko and G.J. Woeginger, A Solvable Case of the Quadratic Assignment Problem, *Operations Research Letters*, 22: 13-17, 1998.

[61] M.G.C. Resende, K.G. Ramakrishnan, and Z. Drezner. Computing Lower Bounds for the Quadratic Assignment Problem With An Interior Point Algorithm For Linear Programming, *Operations Research*, (43): 781-791, 1995.

[62] B. Mans, T. Mautor and C. Roucairol. A Parallel Depth First Search Branch and Bound for the Quadratic Assignment Problem, *European Journal of Operational Research*, 81(3): 617-628, 1995.

[63] P.M. Hahn, T.L. Grant, and N. Hall. A Branch-And-Bound Algorithm For The Quadratic Assignment Problem Based On The Hungarian Method, *European Journal of Operational Research*, 108: 629-640, 1998.

[64] F. Glover. Tabu Search Part 1. *Orsa Journal on Computing* 1, 3: 190-206, 1989.

[65] E. Angel and V. Zissimopoulos. On The Quality Of Local Search For The Quadratic Assignment Problem. *Discrete applied mathematics.* 82: 15-25, 1998.

[66] E. D. Taillard. Robust Taboo Search for The Quadratic Assignment Problem. *Parallel Computing* 17: 443-455, 1991.

[67] S. Kirpatric, C.D. Gelatti and M.P. Vecchi. Optimization By Simulated Annealing. *Science*, 220: 671-680, 1983.

[68] M.R. Wilhelm and T.R. Ward. Solving Quadratic Assignment Problems By Simulated Annealing. *IIE Transactions*, 19 (1): 107-119, 1987.

[69] P. Tian, H. Wang, D. Zhang. Simulated Annealing For The Quadratic Assignment Problem: A Further Study, *Computers & Industrial Engineering*, 31 (3-4): 925-928, 1995.

[70] D.T. Connolly. An Improved Annealing Scheme For The QAP, *European Journal of Operational Research*, 46: 93-100, 1990.

[71] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi. Optimization By Simulated Annealing, *Science,* 220(4598): 671-680, 1983.

[72] D.M. Tate and A.E. Smith. A Genetic Approach to The Quadratic Assignment Problem, *Computers and Operations Research*, 22 (1): 73-83, 1995.

[73] R.K. Ahuja, J.B. Orlin and A. Tiwari. A Greedy Genetic Algorithm for The Quadratic Assignment Problem. *Computers and Operations Research* 27: 917-934, 2000.

[74] Gu, Yu and Chung, Christopher A. Genetic Algorithm Approach To The Aircraft Gate Reassignment Problem. *ASCE Journal of Transportation Engineering*, 125(5): 384-389, 1999.

[75] M.O. Ball, B.K. Kaku and A. Vakhutinsky. Network-Based Formulations of the Quadratic Assignment Problem. *European journal of operational research,* 104: 241-249, 1998.

[76] V. Nissen. Solving the Quadratic Assignment Problem with Clues from Nature. *IEEE Transactions on Neural Networks*, 5 (1): 66-72, 1994.

[77] C.A. Oliveira, P.M. Pardalos, and M.G.C. Resende. GRASP With Path-Relinking For The QAP. *Proceedings of the Fifth Metaheuristics International Conference (MIC2003)*, 57: 1-6, 2003.

[78] L. M. Gambardella, E. Taillard, and M. Dorigo. Ant Colonies For The Quadratic Assignment Problem. *Journal of the Operational Research Society*, 50: 167 -176, 1999.

[79] J.P. Braaksma and J.H. Shortreed .Improving Airport Gate Usage with Critical Path, *Transportation Engineering Journal*, 97 (2): 187-203, 1971.

[80] K. Tsuchiya and Y. Takefuji. A Near Optimum Parallel Algorithm For One-Dimensional Gate Assignment Problems. *Proceedings of 1993 International joint Conference on Neural Networks.* 1993.

[81] R.P. Brazile and K.M. Swigger. Generalized Heuristics For The Gate Assignment Problem. *Control and Computers.* 19 (1): 27-32, 1991.

[82] P. Merz and B. Freisleben. A Comparison Of Memetic Algorithms, Tabu Search, And Ant Colonies For The Quadratic Assignment Problem, *Proceedings of the 1999 International Congress of Evolutionary Computation, IEEE Press, New York:* 2063-2070, 1999.

[83] P. Pardalos, K. Murthy and T. Harrison. A Computational Comparison Of Local Search Heuristics For Solving Quadratic Assignment Problems, *Informatica* 4 (1-2): 172-187, 1993.

[84] V. Maniezzo, M. Dorigo and A. Colorni. Algodesk: An Experimental Comparison Of Eight Evolutionary Heuristics Applied To The QAP Problem. *European Journal of Operational Research*, 81: 188-204, 1995.

[85] E. Nugent, T.E. Vollman, and J. Ruml. An Experimental Comparison of Techniques for the Assignment Of Facillities To Locations, *Operations Research*, 16(1): 150-173, 1968.

[86] R. Bermudez and M.H. Cole. A Genetic Algorithm Approach To LTL Breakbulk Terminal Door Assignment, *Proceedings of the 2000 Industrial Engineering Research Conference, Cleveland, Ohio*, 2000.

[87] XPRESS. http://www.dashoptimization.com.

[88] Rockwell Arena Simulation. http://www.arenasimulation.com.

[89] Watkins Motor Lines. http://www.watkins.com/.

[90] A. Yalcin, P. Deshpande, J. Zayas-Castro, L. Herrera, Simulating and LTL Terminal, *International Journal of Benchmarking*, to appear.

APPENDICES

Appendix A. Xpress Code for Example 2

```
Model 4-dock-12-Trailer
uses "mmxprs","mmquad"
declarations
        NT=4    ! Number of trailers
        ND=2  ! Number of docks
! Defining Sets
        T = 2..NT+1       ! Set of trailers 1
        TT= 2..NT+2       ! Set of trailers 2
        TTT= 1..NT+2         ! Set of trailers 3
        D = 1..ND          ! Set of docks
! Defining Vectors
        EARLIEST: array(D) of real ! Start of time windows
        LATEST: array(D) of real ! End of time windows
        A: array(TTT) of real      ! Planned arrival times
        G: array(TTT) of real          ! Planned service times
        S: array(TT,D) of mpvar    ! Slack between trailers
        X: array(TTT,D) of mpvar   ! Binary variable X
        Y: array(TTT,TT,D) of mpvar ! Binary variable Y
        E: array(TTT,D) of mpvar   ! Enter time of trailers
        L: array(TTT,D) of mpvar   ! Leave time of trailers
        H: integer ! Very large integer
        z: qexp ! Quadratic Objective Function
end-declarations
! User Input
        A:= [0,0,1,1,1,3,3,4,5,6,8,8,9,12] ! Planned arrival times
        G:= [0,1,2,4,2,3,4,4,5,2,1,3,2,0] ! Planned service times
        H:=1000 ! Very large integer
        EARLIEST:= [0,0]  ! Start of time windows
        LATEST:= [10,10]   ! End of time windows
! Constraint 1
        forall(j in T) SUM(k in G) X(j,k) = 1
! Constraint 2
        forall(j in T,k in D) E(j,k) >= A(j) * X(j,k)
```

Appendix A. (Continued)

```
! Constraint 3:
        forall(j in T,k in D) L(j,k) - G(j) * X(j,k) = E(j,k)
! Constraint 4
        forall(j,i in T,k in D| i<j)  E(j,k) >= L(i,k) + H*(X(i,k)+X(j,k) - 2)
! Constraint 5
        forall(j in T,k in D) L(j,k) <= LATEST(k)
! Constraint 6
        forall(i in TTT,j in TT,k in D | i<j) Y(i,j,k) <= (X(j,k) +X(i,k)) / 2
! Constraint 7
        forall(j in TT,k in D) SUM(i in 1..j-1) Y(i,j,k) = 1
! Constraint 8
        forall(i in TTT,j in TT,k in D | i<j) S(j,k) >= E(j,k) - L(i,k) + H*(Y(i,j,k) - 1)
! Constraint 9
        forall(j in T,k in D) X(j,k) is_binary
! Constraint 10
        forall(i in TTT,j in TT,k in D | i<j) Y(i,j,k) is_binary
! Initializations
        forall(k in D) E(NT+2,k) = LATEST(k)
        forall(k in D) X(NT+2,k) = 1
        forall(k in D) L(NT+2,k) = LATEST(k)
        forall(k in D) E(1,k) = EARLIEST(k)
        forall(k in D) L(1,k) = EARLIEST(k)
! Objective Function
        z:= SUM(j in TT)SUM(k in D) S(j,k)^2
        minimize(z)
! Output
        writeln("Objective Function: ", getobjval)
        forall(j in 2..NT+2,k in G )
        writeln("S",j,k,"-","Value:--->", getsol(S(j,k)))
        forall(j in 1..NT+2,k in G ) do
        if getsol(X(j,k)) <> 0 then
        writeln("______","X",j,k,"-","Value:--->", getsol(X(j,k)))
        end-if-do-model
```