12-2010

ORDER PICKING AND STORAGE USING STACKABLE PALLETS IN A WAREHOUSE

Ahmed Hassan aly
Clemson University, ahassan@g.clemson.edu

ORDER PICKING AND STORAGE USING STACKABLE PALLETS IN A WAREHOUSE

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Industrial Engineering

---

by
Ahmed Ibrahim Hassan Aly
December 2010

William G. Ferrell, Jr., Committee Chair
Scott A. Shappell
Kevin M. Taaffe
Maria E. Mayorga

ABSTRACT

Quarter Billion dollars could be saved annually by double stacking pallets. A forklift storing 100 double stacked pallets saves 2.5 working hours versus single pallets. More than one billion pallets and cases handled between Wal-Mart distribution centers according to BEA White Paper (2006) [1], assume only 500 million were stackable pallets. This translates to 12.5 million working hours. If a forklift driver earns $20 per hour, then $250 million can be saved.

Moreover, handling double stacked pallets takes up to 46 % of total time to pick and store. This is a significant element that was ignored in literature. All previous research focused only on travel time that is just part of the total time.

In this dissertation we model dual command operations to find optimal path with minimum travel time. Combination of both picking and storage activities is also known as dual command. Our environment is a manual warehouse where full pallets can be double stacked. Accordingly, three time-based models were developed as 123 steps to reach dual command model; order picking model, storage model and combined storage and order picking model. The mathematical models find the optimal sequence of storing and picking pallets that leads to the minimum travel time. Using those models allow double stacking pallets, therefore around half of material handling time, labor cost, and equipment cost will be saved compared to single stacked pallet operations. Two heuristics were also developed that gave sub optimal but quicker solutions that can also be used by the warehouse management and reduce the travel time significantly as well.

DEDICATION

I dedicate this work to Allah (God) who created me and created every human being. I thank Allah (God) for making it easy for me to complete this dissertation.

ACKNOWLEDGMENTS

First of all, I thank Allah (God) who is the reason behind my existence and success in this research. Allah (God) says in the Quran, chapter 13, verse 29: “For those who believe and work righteousness, is (every) blessedness, and a beautiful place of (final) return.” Allah (God) says in the Quran, chapter 3, verse 146: “And Allah loves those who are patient.” Allah (God) says in the Quran, chapter 94, verses 5 and 6: “So, verily, with every difficulty, there is ease, Verily, with every difficulty there is ease.” I thank prophet Muhammad peace be upon him (pbuh) and peace be upon all the prophets of God. Prophet Muhammad (pbuh) did his best to convey the religion of Islam which teaches 2 main things: worship one God and be good to fellow humans. The teachings of Islam helped me in being a better human and proceed through the test of life with an optimistic attitude. I thank my wife Yara Ibrahim, who was always beside me helping me throughout the PhD journey. She really deserves her name to put on this degree beside my name.

I thank my Advisor Dr. William Ferrell who did a great effort in guiding me throughout this research. May God bless you, and reward you for your sincere help and support. I thank my Advisory committee, Dr. Scott Shappell, Dr. Kevin Taaffe and Dr. Maria Mayorga. I like to thank my father Dr. Ibrahim Hassan who encouraged me to pursue the PhD. I thank my mother Eman Mashaly for her prayers and great emotional support. I thank my mother in law Hamida Tawfik for her prayers, and father in law Abdelmoneim Ibrahim, brother Mohamed and sister Heba who always supported me in my PhD journey. I thank our department chair, Dr. Anand Gramopadhye and all Clemson university staff, especially Ms. Saundra Holland, Ms. Beverly Robinson and Mr. Martin Clark for their help and for making us feel like one family. I thank all my professors for their openness and all knowledge they passed to me. I thank everybody who helped me in one way or another during my PhD journey. I thank my brothers at the Islamic society of Clemson, especially Dr. Tanju Karanfil, Imam Mustafa Khattab, Dr. Abdul Baasit Shaibu, Dr. Mohammed Salah, Dr. Abdul Aziz Khan, Brother Nedal Mefleh, Brother Mooez Masood, Brother Nadeem Vellore, Brother Yousef Qaroush and Brother Mohammed Darfoon.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

This research is focused on providing a contribution on a specific gap; the state of the art associated with picking and storage in manual warehouses. The order picking problem is a well known problem in the warehouse literature. Order picking is the process of retrieving items on a pick list from their storage locations. Items can range from small cases (less than pallet size) that have to be picked manually or an automated storage retrieval system AS/RS up to fully loaded pallets that are picked by material handling equipment such as a forklift. In this research focus will be on fully loaded pallets. Rick LeBlanc states that: "High lift trucks made possible vertical stacking of unit loads and a resulting dramatic improvement of warehouse and plant storage efficiencies" [2]. Vertical stacking of unit loads is addressed in this research by considering double stacking of fully loaded pallets. The warehouse considered in this research has activities such as receiving, storage (also known as put away process), order picking, packing and shipping. Literature on order picking focused on using graph theory in solving this problem that does not consider the time element. Few mathematical models were developed in the literature for the order picking problem but they did not consider the double stacking of pallets and all the time elements associated with pallet handling. This motivated me to pursue this research that focuses on order picking and storage in manual warehouses. Moreover, the research addresses fully loaded pallets that can be double stacked while being transported by a material handling device like a forklift.

The research contributions are developing a methodology to include stackability in determining the optimal pick and storage sequence, and developing a mathematical programming approach to this problem that is time-based rather that distance-based so the critical tasks associated with stacking can be correctly included in the model. Furthermore, developing a detailed time-based math programming model for the dual command problem (allows picking and storing in the same route), and developing two heuristics for the dual command problem. In addition, showing that pallet handling time ranges from 22% to 46% of the total time to pick and store pallets. This time element was not addressed in literature, and rather travel time was the main focus.

This dissertation report is arranged as follows:

-Chapter 2 is a literature review of the relevant research done in this area.

-Chapter 3 establishes the methodology for modeling stackability and presents two time-based mathematical models, one for order picking and one for storage.

-Chapter 4 is a paper that will be submitted to a journal. This paper combines both picking and storage activities, also known as dual command into a single time-based mathematical model. Also, this paper presents two heuristics that provide quick sub optimal solutions to the same dual command problem.

-Chapter 5 presents conclusions from this interesting research.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter provides an overview of the literature related to all research reports in this document; that is, the literature associated with Chapter 3 and Chapter 4. Since, the literature on order picking involves hundreds if not more than a thousand papers in its entirety; we focused in this chapter only on the relevant papers. For example, the bibliography developed by Goetschalckx and Wei [3] contains 413 papers; however, many of these papers address problems that are not relevant to this research. To illustrate, research on order picking with automated storage retrieval systems (AS/RS) has a different technical focus so it is not included. The same is true for some of the work that adapts classic operations research problems like the vehicle routing problem. The chapter is divided into three main sections: order picking, pallet storage, and combined storage and picking.

2.2 Order picking

When orders are received from customers, a “pick list” is generated that is simply the aggregated individual items required to fill these orders. Order picking is the process of retrieving items on a pick list from their storage locations. Items can range from small cases (less than pallet size) up to fully loaded pallets. Small cases are either picked by humans or reside in an AS/RS, while fully loaded pallets are picked by material handling equipment such as a forklift. Bartholdi and Hackman [4] estimated that order picking accounts for 55% of the total operational warehouse costs, while Coyle at al. [5] estimated that order picking accounts for 65% of the total operational warehouse costs. This means order picking is the biggest cost element in the warehouse operational costs. Literature on order picking seeks to minimize the cost, measured in a variety of ways, associated with retrieving the items on a pick list.

Literature reviews have been performed exclusively on this topic. For instance, Gu and et al. [6] published a comprehensive literature review of research on warehouse operations that included picking along with other functions. Besides, De Koster, et al. [7] looked at this general topic whereas van den Berg and Zijm [8] focused on order picking models and problem classifications.

In this research, the focus is exclusively on pallets that are handled by forklifts so the remainder of this chapter will be confined to this subset of warehouse operations and not more general topics. To demonstrate, there is one paper that addressed stackability, however for a different problem. Steudel [9] developed an algorithm that solved the common problem of loading rectangular items on a rectangular pallet using the option of stacking items on their end and/or side surface to maximize the number of items per layer on the pallet deck board. In Steudel's research, case picking was considered where cases (boxes) are stacked on top of each other on a single pallet. In our research we assume two full single pallets are stackable on each other.

The literature relevant to full pallet picking naturally divides into two categories based on the measure that the researcher chooses to make the pick route efficient. Some researchers minimize the total distance traveled while others minimize the total time to pick.

2.2.1 Distance-based models for order picking

This section sheds light on the different modeling and solution approaches that researchers used to solve the order picking problem. Earlier, graph theory was used where nodes represent locations of items to be picked in a warehouse and arcs represent the possible routes the forklift could travel during picking. In 1983, Ratliff and Rosenthal [10] were the first to use this approach for the order picking problem. The objective function is minimizing the distance traveled by the forklift while picking ordered items stored in a warehouse and transporting them back to the shipping area. Their research was restricted to order picking in a rectangular warehouse that contains crossovers (i.e. pass ways between main aisles) only at the ends of aisles. In 1988, Goetschalckx and Ratliff [11] used graph theory to develop a minimum distance algorithm for routing order pickers in a warehouse with one cross aisle. However, they considered picking items less than a pallet size.

Later, traveling salesman problem (TSP) was used to model the order picking problem. In 1998, Daniels et al. [15] modeled warehouse order picking as a traveling salesman problem. The model developed by Daniels et al. [15] determined the sequence of visiting the unique locations where each part in the order is stored. In addition, the model developed by Daniels et al. [15] minimized the total cost associated with both the assignment of inventory to an order and the associated sequence of visiting the selected locations. However, Daniels et al. [15] addressed cost as arbitrary cost of moving from one location to another having only one mathematical term in the objective function. Indeed, this cost element is too general as Daniels et al. [15] focused in their paper on assignment of inventory to an order. Yet, in my research the objective function is broken down into different unique sub cost elements, each associated with either a travel time element or pallet handling time element. Moreover, cost is the driving factor in my research where in every move that a forklift can make is modeled as a unique separate cost element in the objective function.

Another TSP paper, in 2010, Theys et al. [16] developed a traveling salesman problem heuristic for routing order pickers in warehouses using distance as their evaluation criteria. However, in our research we preferred not to use the TSP approach as we look at the time not the distance traveled by order picker to be able to include the different time elements associated with handling and stacking pallets.

Simultaneously, in the literature, the order picking problem was addressed by linking it to storage and routing strategies to be implemented in a warehouse. As an illustration, in 1993 Hall [17] evaluated and compared strategies for routing a manual picker through a simple warehouse. Hall [17] derived rules of thumb for selection of order picking strategies and optimization of warehouse shape. Our research assumes warehouse shape or layout is given and focuses on generating a new strategy for each picking problem according to which pallets are stackable on each other.

Later in 1998, Caron et al. [18] calculated expected travel distance for different routing strategies namely traversal and return policies in low-level picker to part systems. Caron et al. [18] considered distance, meanwhile, in our research we considered time which is more comprehensive than distance in addressing not only the travel moves of the forklift but also the pallet handling moves.

Recently, some papers address the order picking problem in relation to the order batching problem. For example, in 2008 Tsai et al. [19] developed an algorithm that searches for the most effective travel path for a batch by minimizing the travel distance. However Tsai et al. [19] used a distance approach, while in this research the more comprehensive time approach was used.

Probabilistic picking model is another solution approach in literature. In 2005, Le Duc and De Koster [20] developed a probabilistic model that estimates the average travel distance of a picking tour; however their model focused on finding the optimal zoning scheme with respect to minimizing the average travel distance. In 2006, Hwang and Cho [21] developed a performance evaluation model for the order picking warehouse system in supply center (SC) by reducing the travel distance of transporters. Hwang and Cho [21] used probabilistic picking while we use deterministic picking in our research. We used deterministic picking as our concern is to reduce the time spent by the forklift driver while picking all items in an order that is requested by a customer with no probability involved with picking each item.

Some researchers integrated the warehouse layout problem with the picking problem. For instance, in 1992, Gray et al. [22] modeled in general terms the composite design and operating problems for a typical order-consolidation warehouse. Gray et al. [22] addressed warehouse layout, equipment and technology selection, item location, zoning, picker routing, pick list generation and order batching. In 2006, Roodbergen and

Vis [23] evaluated the relationship between the layout of the order picking area and the average length of a picking route for areas consisting of one block. In our research we focus on the order picking problem in a given warehouse layout that has cross aisles. These cross aisles allow forklift driver to reach pallets located in different aisles in shorter time.

Finally, researchers used travel distance based algorithm for solving the order picking problem. To illustrate, in 1990, Rana [24] developed a travel distance based algorithm for order picking in narrow-aisle warehouses, where one forklift can travel in one direction in the aisle. Another example, in 2007, Manzini et al [25] introduced an analytical model and a multi-parametric dynamic model to quickly estimate the travelled distance during a picking cycle. Nevertheless, in our research we look at the time traveled by order picker not only the distance.

2.2.2 Time-based models for order picking

The literature we discussed in the previous section on distance models in order picking was more coherent, organized and developed as researcher's efforts were made building on each other and extending the work of each other. However the literature in this section on time models and the next sections is more fragmented as efforts were made in different areas that sometimes cannot be linked together.

Before we proceed, let's define two keywords in literature. First, single command; it stands for performing either storage or picking in one trip by the forklift driver in the warehouse. Second, dual command; it stands for combining storage and picking in one trip by the forklift driver. In dual command, workers travel loaded from the pickup and deposit (P&D) point first to a location to store a pallet, then to a second location to pick a pallet and then return to the P&D point. We name the P&D point in our research depot.

There are several time-based approaches to the picking problem. For instance, in 1993, Hwang and Song [26] developed the expected travel time models based on the probabilistic analysis for single and dual commands assuming randomized storage assignment policy. In our research, we don't use probability when dealing with single and dual commands as we let the model choose the strategy (single or dual) that would lead to shorter picking time. Another paper, in 1998, De Koster and Van der Poort [12] used graph theory as time based solution approach and extended the work in [10] and [11]. De Koster and Van der Poort [12] addressed warehouses with a central depot and decentralized depositing. Decentralized depositing allows order picking trucks (forklifts) to pick up and deposit pallets at the head of every aisle without returning to the depot. Furthermore, in 2001, Roodbergen and De Koster [14] used graph theory as time based solution approach and extended the work by De Koster and Van der Poort [12]. Roodbergen and De Koster [14] included parallel aisle warehouses where order pickers can change aisles at the end of every aisle and also at a cross aisle halfway along the aisles. Moreover, in another paper, Roodbergen and De Koster [13] used graph theory as solution approach and constructed a dynamic programming algorithm for calculating order picking tours of minimal length in warehouses with up to three cross aisles. These five papers [10], [11], [12], [13], [14] utilized a graph theory approach because the objective is strictly minimizing either distance or time. In our research we look at the time traveled by order picker not only the distance to include the different time elements associated with handling and stacking pallets, that is the gap that is addressed in this research.

Some researchers used queuing theory to solve order picking problems. To illustrate, in 1999, Chew and Tang [27] modeled order picking systems as queuing systems and validated the results with simulation. Later, in 2007, Le-Duc and De Koster [28] extended the work by Chew and Tang [27] addressing the order batching problem in a two-block warehouse when less than full pallets are picked. Moreover, in 2009, Yu and De Koster [29] constructed an approximation model to analyze the impact of order batching and picking area zoning on the mean throughput time in an a pick-and-pass order picking system using queuing theory. Finally, in 2010, Parikh and Meller [30] developed analytical expressions for expected travel time that was used in a math program to optimize warehouse design parameters. Parikh and Meller [30] determined the optimal storage system configuration such as the height of the storage aisles. All of the research efforts above that developed time models did not look into handling stackable pallets by forklift during picking. This research is unique as it addresses the conventional order picking problem, however it introduces the new feature of the forklift carrying two stackable pallets at the same time with all the time elements associated with it. Stackable pallets add the complexity of time elements related to handling those two pallets. For instance, if a forklift driver is carrying one pallet and desires to pick a second pallet, time elements involved are unloading one pallet on the ground, picking the second pallet from the rack, and double stacking them together. Moreover, using stackable pallets cuts the total picking time by half as it cuts the number of trips to pick a certain number of pallets by half. This is achieved if you can double stack two pallets together and save the second trip of picking the second pallet singly. Double stacking pallets while picking saves half the material handling time, which is the gap in literature that my work fills.

2.3. Pallet Storage

Storage is the process of assigning items to their storage locations in a warehouse. Most of the storage models address automated warehousing that is not the topic of our research as we focus here on manual warehousing. On the other hand, in manual warehousing, for instance Queirolo et al. [31] developed a simulation model that determines where to assign storage areas in a warehouse. Queirolo et al. [31] developed a model that reduces the global storage cost through minimizing the total travel time. However, Queirolo et al. [31] addressed a different problem that is the warehouse layout optimization problem while in our research we address the optimization of the storage sequence in a warehouse using stackable pallets.

2.4 Combined Storage and Picking

Most of the models developed for combining storage and picking addressed automated warehousing such as automated storage and retrieval systems (AS/RS). AS/RS is not the topic of our research as we focus here on manual warehousing. However there is some terminology that we can learn from AS/RS that we use in this research. To illustrate, in 1977, Graves, et al. [32] used storage-retrieval interleaving in automatic warehousing systems. Interleaving is the sequencing of storage and retrieval requests. We apply interleaving in our research when we combine storage and picking in one trip in the dual command problem. Another example, in 1993, Eynan and Rosenblatt [33] stated that "in order to improve their service level many automated storage/retrieval systems (AS/RS) have adopted a dual command policy. In a dual command policy both storage and retrieval of pallets are done within one roundtrip of the crane." This supports the dual command approach that we used in this research however we used it for manual warehouses. Moreover, in 1997, Lee and Schaefer [34] sequenced storage and retrieval requests using unit-load AS/RS applications. In our research we also consider unit load that is fully loaded pallets however for we use it in manual warehouses. Besides, Lee and Schaefer [34] stated that "We find that the sequencing methods can significantly reduce travel time by a storage and retrieval machine, thereby, increasing throughput." This statement supports the sequencing approach we use in this research when we combined storage and picking.

Finally, in 2009 Pohl et al. [35] developed an expression for expected travel distance for dual command operations. Pohl et al. [35] used the expression for expected travel distance to analyze three common warehouse designs. Pohl et al. [35] concluded that warehouse design layout C in their paper is the best layout. Layout C had racks parallel to the shipping dock with aisles perpendicular to the shipping dock. Pohl et al. [35] conclusion confirmed our choice of this warehouse layout C for our problem, besides it was the same layout that I have seen at the Welcome road external warehouse at Robert Bosch Anderson plant in South Carolina.

2.5 Other papers

Our research can be also viewed as routing problem where we route the forklift in the warehouse while picking and storing to minimize total time of both handling pallets and traveling. Therefore, straddle carrier routing problem in a container terminal is similar to our problem however it minimizes the total travel time of the straddle carrier without looking at the handling time of the containers. A sample paper from this area is the one by K.Y. Kim and K.H. Kim [36] in 1999. K.Y. Kim and K.H. Kim [36] developed a routing algorithm for a single straddle carrier to load export containers onto a containership.

Here are some papers that are not referenced in the dissertation, but are helpful: [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], and [56].

2.6 Conclusion

In conclusion, previous research has established the need for applying rigorous approaches to the picking and storage problems. Distance based models have been the focus of researchers attention for over 27 years with graph theory as a commonly used foundation. Time based approaches are much more recent and have focused on total time traveled in the warehouse during picking. Our literature review concluded that our research is different from all of the aforementioned because it considers time elements involved with double stacking full pallets in a manual warehouse, that is the gap in literature that my work fills.

CHAPTER THREE

ORDER PICKING AND STORAGE WITH STACKABLE PALLETS

Abstract

Order picking and storage are the main activities in a warehouse. In recent years, pallets have been designed that are stackable on each other which create an interesting research problem that is explored here. While the objective remains to sequence picking or storage in the most efficient way, the approach required for including this ability to stack pallets is different. A stackability matrix is proposed that identifies the pallets that can be stacked and allows this feature to be included in the model. Also, a potentially significant amount of time is consumed when pallets are stacked so a time-based mathematical programming approach is needed. Three models are presented here that add these features and numerical examples are included to illustrate how the model can be used in more practical situations.

Keywords

Order picking, order storage, path optimization, pallet picking, travel time, pallet stackability

3.1 Introduction

This research centers on the new feature of stacking pallets that is becoming more common in traditional warehouses that store and retrieve raw materials, semi-finished products and/or finished products on demand. Activities in such a warehouse are well known and can be divided into 5 categories: receiving, order storage (also known as the put away process), storage, order picking, and packing and shipping. The focus of this work is on order picking and order storage in manual warehouses that only handle full pallets with the unique feature that some of the pallets can be stacked on each other while being transported by the material handling device. The goal is to determine the sequence in which the pallets should be picked or stored and the route for the most efficient operations given that pallets can be stacked.

Determining pick paths is certainly not a new problem; however, we submit that an important consequence of including stacking is that the objective function must change from minimizing distance to minimizing time. When pallets are not stacked, it seems reasonable that these two objectives would produce identical routes; however, when stacking is allowed this is not necessarily true. For simplicity, this research will refer to the material handling device used in the warehouse as a forklift. To store two pallets in the same trip, a forklift must first double stack the two pallets, move them to the first pallet storage location, put them down on the floor, pick up the pallet to be placed in the rack, store it, pick up the second pallet and then proceed to store it in the correct location. As such, it is proposed that the correct objective function for the picking and storage problem with stackable pallets is minimum time and the approach to achieve this is to use mathematical programming. The warehouse layout used in this paper is the warehouse with cross aisles.

3.2 Literature Review

The literature in this general research area is quite extensive so only a few key papers that are most directly related to this research are referenced here. We acknowledge previous work in three important areas: Order picking, distance based approaches and time based approaches.

3.2.1 Order picking

Steudel [9] developed an algorithm which solved the common problem of storage rectangular items on a rectangular pallet using the option of stacking items on their end and/or side surface to maximize the number of items per layer on the pallet deck board. Steudel considered case picking where cases are stacked on top of each other, while in our research pallet picking is considered allowing full pallets to be stacked on top of each other, while in Steudel research focus was on the loading pattern of smaller items (cases) on a single pallet.

3.2.2 Distance-based approaches

Ratliff and Rosenthal [10] is one of the early papers that address the order picking problem in a rectangular warehouse that contains crossovers only at the ends of aisles. De Koster and Van der Poort [10] extend the Ratliff and Rosenthal algorithm to include warehouses with a central depot and also allowed order picking trucks to pick up and deposit pallets at the head of every aisle without returning to the depot. Roodbergen and De Koster [13] extend this order picking situation to include parallel aisle warehouses with two or more cross aisles. De Koster and Van der Poort [10] address the problem of finding the shortest order picking routes in a warehouse with either a central depot or with decentralized depositing (i.e., order picking trucks can pick up and deposit pallets at the head of every aisle without returning to the depot). An interesting aspect of this paper is that the authors exploit an observation that modern warehouses are becoming paperless by using mobile devices to convey pick lists that contain picking locations to order pickers instead of collecting them from a central printer. These three papers utilize a graph theoretic approach because the objective is strictly minimizing distance. Pohl et al. [35] developed an expression for expected travel distance for dual command operations. In our research we modeled the actual travel distance for dual command operations not just the expected distance. Moreover in our research, the distance model is just a first step towards developing the time model which is the real contribution of this paper.

3.2.3 Time-based approaches

There are several time-based approaches to the picking problem. Le-Duc and De Koster [28] modeled order picking systems as queuing systems and validated the results with simulation for an order batching process in a two-block warehouse when less than full pallets are picked. Chew and Tang [27] also modeled order picking systems as queuing systems and validated the results with simulation to analyze order batching and storage allocation strategies in an order picking system. Yu and De Koster [29] construct an approximation model to analyze the impact of order batching and picking area zoning on the mean throughput time in a pick-and-pass order picking system using queuing theory. Parikh and Meller [30] developed analytical expressions for expected travel time, and then they use these expressions in a math program to optimize warehouse design parameters. Their research determines the optimal storage system configuration such as the height of the storage aisles.

In conclusion, previous research has established the need for applying rigorous approaches to the picking and storage problems. Distance based models have been the focus of researchers attention for over 25 years with graph theory as a commonly used foundation. Time based approaches are much more recent and have focused on total time traveled in the warehouse during picking. Our literature review only identified one paper that addressed stackability within a single pallet for cases which are less than a pallet size. These concluded that this research is different from all of the aforementioned because it considers stackability for full pallets and also considers time elements involved with stackability.

3.3 The Models

The basic research problem is to determine the minimum time pick paths for a manual warehouse when pallets are allowed to be stacked on each other. Figure 3.1 illustrates an example of this order picking process. A forklift starts from point 1 at the depot, picks the pallet at point 3 (passing by point 2), and then point 4 to double stack the pallet that was at 3 on the one at 4. The forklift then returns to the depot at point 5 with the two double stacked pallets. A warehouse layout with cross aisles allows the forklift driver to reach pallets in a shorter distance compared to a warehouse with no cross aisles.

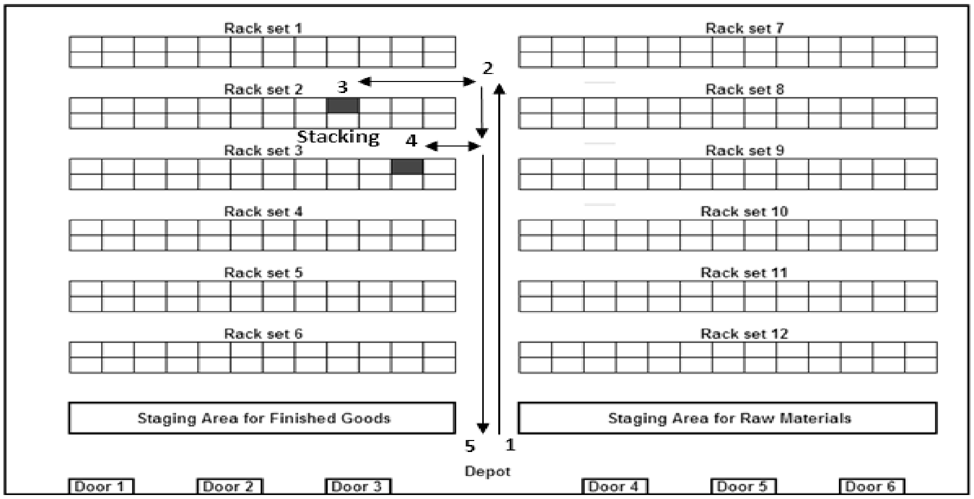


Figure 3.1: Warehouse layout

Modeling this situation requires a time-based mathematical programming approach and the ability to include stackability in the model. The latter is achieved by creation of a stackability matrix that identifies the pallets that can be stacked on each other. If there are *n* pallets to be picked, the stackability matrix is simply an *n+2* x *n+2* matrix where element (i, j) equals 1 if pallet i is stackable on pallet j and 0 otherwise. The two extra locations are for dummy and depot which have 0 elements with all other pallets. Note that this matrix is not necessarily symmetric across the diagonal since a pallet of glasses could be stacked on a pallet containing sheets of carbon steel but not vice versa. A new measure we will call stackabilty density is introduced, which is the % of possible stacking options available. For example, in an 8 pallet example, the stackability matrix can allow all pallets to be stackable on each other if we had 56 ones in the matrix which is equivalent to stackability density of 100%. This means for example, if we had only 8 ones in the matrix, the stackability density is equal to 14.29 % which is 8 divided by 56; the maximum number of one's allowing all pallets to be stackable on each other except on itself.

3.3.1 Including stackabilty

Hassan and Ferrell [57] included stackabilty by developing a Boolean programming model to minimize total travel distance that allowed multiple picks in any or all routes. The model utilizes the following assumptions:

- One forklift is performing the order picking function and each route begins and ends at the depot.
- There is a one-to-one mapping between locations and pallets.
- The time spent during picking and stacking is zero, as an initial step where time will be considered in the time based picking model in the next section.
- Whether pallet i is stackable on pallet j or vice versa, the stacking can be accomplished at either location.
- A maximum of two pallets may be stacked.

To construct this model, let:

- S_{ij} = 1 if pallet i is stackable on j; 0 otherwise (input parameter)
- c_{ij} = the distance between pallet i and pallet j (input parameter)
- X_{ij} = 1 if the path includes moving from node i to node j; 0 otherwise (decision variable)

- i,j = indices that identify the pallet/location; depot is 1, pallets/locations to be picked are 2 through n+1; dummy node is n+2 and required for model completeness (Note: $c_{i,n+2}=0\ \forall$ i)

A dummy node is required because the general model must allow for *n* single picks; however, when stacking is found in the optimal solution the number of routes is less than *n* so the model must contain a zero time option to accommodate these "extra" routes.

The model from [57] that includes stackability is:

Minimize $Z = \sum_{ij} c_{ij} * X_{ij}$

Subject to

$$\sum_{i=2}^{n+2} X_{1i} \geq 1 \tag{1}$$

$$X_{ij} + X_{ji} \leq S_{ij} + S_{ji}\text{, i = 2, 3… n+2, j = 2, 3… n+2} \tag{2}$$

$$\sum_{i=1}^{n+2} X_{ji} \geq X_{1j}\text{, j = 2, 3… n+2} \tag{3}$$

$$X_{n+2,1} = X_{1,n+2} \tag{4}$$

$$X_{j1} \geq \sum_{i=2}^{n+2} X_{ij}\text{, j = 2, 3… n+2} \tag{5}$$

$$\sum_{i=1}^{n+2} X_{ij} = 1\text{, j = 2, 3… n+2} \tag{6}$$

$$\sum_{i=1}^{n+2} X_{ji} = 1\text{, j = 2, 3… n+2} \tag{7}$$

$$\sum_{i=1}^{n+2} X_{ii} = 0 \tag{8}$$

$$X_{ij} + X_{ji} \leq 1\text{, i = 2, 3… n+2, j = 2, 3… n+2} \tag{9}$$

$$X_{ij} = \{0, 1\}\text{, i = 2, 3… n+2, j = 2, 3… n+2} \tag{10}$$

The constraints in this model perform the following functions:

1) Required the first segment of a route to be from the depot (location 1) to a pallet (locations 2 through n+1)
2) Ensures that only pallets in which S_{ij} equals 1 can be stacked. If S_{ij} is equal to 0, X_{ij} is forced to be zero to prevent stacking pallets i and j
3) Requires the second segment of a route is either to a second pallet so that stacking can occur or back to depot
4) Forces return to depot after visiting the dummy node (n+2)
5) Requires that each route returns to the depot after completing a double stack or a single pick
6) Ensures that each pallet is picked once
7) Preserves feasibility by forcing each route to leave a node only once
8) Prevents stacking pallets on themselves
9) Eliminates cycles by requiring that the trip between any two pallets is made only once

3.3.1.1 Numerical Examples

All of the numerical examples in this chapter were solved to optimality using ILOG OPL Development Studio version 5.5 and a Dell personal computer with an Intel Core 2 Duo processor and 2.00 GB of RAM.

The model was generated by placing the middle pallets near the depot and the other pallets incrementally further by 10 distance units each. For example, in 50 pallets example, pallets 25 and 26 are 50 distance units away from depot, while pallets 24 and 27 are 60 distance units further from depot and similarly pallets 10 and 41 are 200 distance units away from depot. See Appendix for distance and stackability appendices.

50 pallets: The model was used to solve a problem with 50 pallets; hence, the stackability matrix is 52x52 although stacking can only occur for 50 locations associated with real pallets. It is assumed that 50 stacking options (chosen randomly) are available so the matrix contains 50 ones and remains entries are zero. The equivalent stackability density for this matrix is 2.04% which is 50 divided by 2450; the maximum number of one's allowing all pallets to be stackable on each other except on itself. Distance matrix was generated based on having 2 middle pallets 25 and 26 closest to depot then all pallets less than 25 and more than 26 are located at increments of 10 from each other. Please see appendix for distance matrix.

The optimal solution for this example is presented below. D represents the depot and Pj represents pallet/location j.

D-P2-P1-D-P3-P4-D-P5-P6-D-P7-P8-D-P9-P10-D-P11-P12-D-P13-P14-D-P15-P16-D-P17-P18-D-P19-P20-D-P21-P22-D-P23-P24-D-P25-P26-D-P27-P28-D-P29-P30-D-P31-P32-D-P33-P34-D-P35-P36-D-P37-P38-D-P39-P40-D- P41-P42-D-P43-P44-D-P45-P46-D-P47-P48-D-P50-P49-D-Dummy-D.

The total distance traveled using this solution is 9750 ft. If only single picks were used, the total distance is computed to be 17000 ft which represent a significant reduction of 7250 ft as expected.

The number of possible pallets allowed to be stacked was set equal to number of pallets. That is, 100 pallet problem has a stackablity matrix that is 102x102 or 10,000 possible opportunities for stacking. In the numerical example, 100 of these 10,000 were selected. Equivalent stackability density for this matrix is 1.01% which is 100 divided by 9,900. The model was generated by placing middle pallets near the depot and other pallets incrementally further by 10 distance units each. For example, in 100 pallets example, pallets 51 and 52 are 50 distance units away from depot, while pallets 50 and 53 are 60 distance units further from depot and similarly pallets 6 and 97 are 500 distance units away from depot. For the 250 pallets example, pallets 126 and 127 are 50 distance units away from depot, while pallets 125 and 128 are 60 distance units further from depot and similarly pallets 31 and 222 are 1000 distance units away from depot. Also, for 250 pallets example, equivalent stackability density for this matrix is 0.4% which is 250 divided by 62,250.

50 more examples: The model has been used to solve problems containing 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 pallets, each was solved using five different variations of stackability options; total of 50 examples. Results are shown below in Figure 3.2. Each data point below is average of 5 replications, each is run with a different stackability matrix, where ones are in the matrix are scattered differently but sum of ones in all 5 different stackability matrices are equal to number of pallets (problem size), where ones are assigned randomly. Distance matrices were similarly generated based on having 2 middle pallets closest to depot then all other pallets were located at increments of 10 from each other.

The computational effort required to find optimal solution increases with problem size as shown in Figure 3.2. This is expected due to the combinatorial nature of underlying problem.

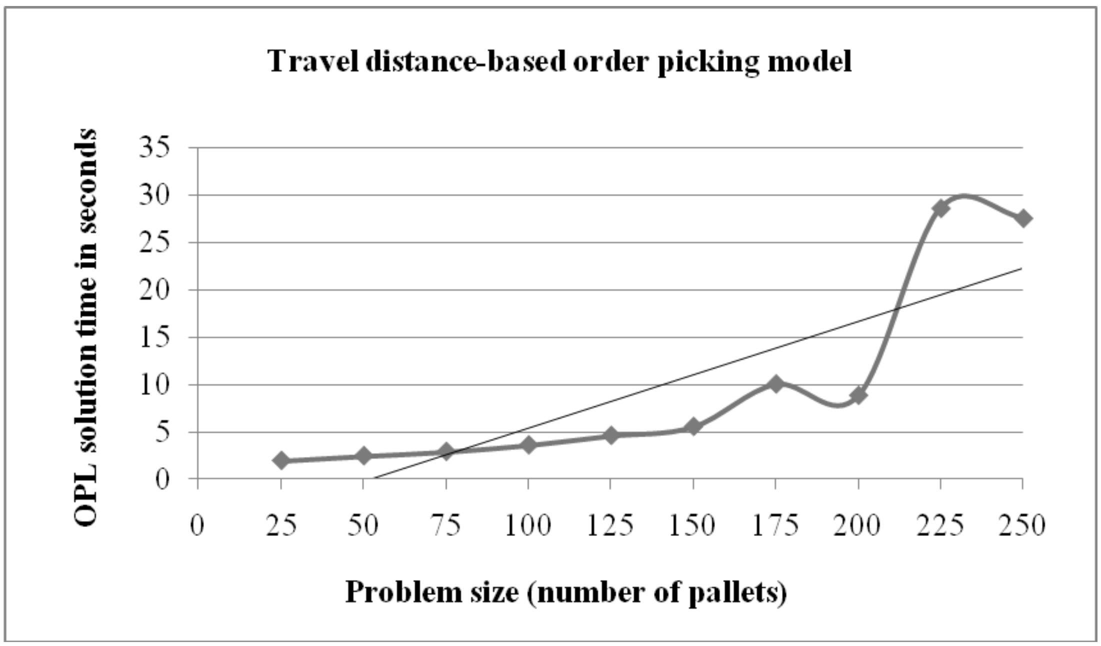


Figure 3.2: Time vs. size in picking distance

3.3.2 Time-based picking model

The previous model was extended to a time-based format that allows inclusion of other tasks associated with stacking pallets that consume time. As before, each pallet is assumed to have a unique location with the depot represented by location 1, the pallets to be picked in locations 2 through n+1, and the dummy location as n+ 2. In addition to S_{ij} and c_{ij}, this model required the following input parameters:

- t_p = time to pick a pallet from rack in minutes
- t_{s} = time to stack one pallet on another in minutes
- s = average speed of the forklift traveling in a warehouse in feet per minute
- d_{ij} = distance from pallet i to pallet j in feet

The decision variables are:

- X_{ij} = 1 if the path includes moving from node i to node j; 0 otherwise
- Y_i = 1 if pallet i is chosen to be picked first; 0 otherwise
- Z_{ij} = 1 if pallet i is chosen to be stacked on top of another pallet j; 0 otherwise
- RD_i = 1 if you return from pallet i to Depot; 0 otherwise

The model that minimizes the total time consumed by the order picker in route and manipulating the pallets is presented below.

$$z = \sum_{k=2}^{n+1} d_{1k} * Y_k / s + \sum_{k=2}^{n+1} \mathrm{t_p} * Y_k + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} d_{kl} * Z_{kl} / s + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} \mathrm{t_p} * Z_{kl} + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} \mathrm{t_s} * Z_{kl} + \sum_{k=2}^{n+1} d_{k1} * RD_k / s$$

Subject to

$$\sum_{i=2}^{n+2} X_{1i} \geq 1 \qquad (1)$$

$$X_{ij} + X_{ji} \leq S_{ij} + S_{ji} \qquad \text{i, j} = 2, 3\ldots \text{n+2} \qquad (2)$$

$$\sum_{i=1}^{n+2} X_{ji} \geq X_{1j} \qquad \text{j} = 2, 3\ldots \text{n+2} \qquad (3)$$

$$X_{n+2,1} = X_{1,n+2} \qquad (4)$$

$$X_{j1} \geq \sum_{i=2}^{n+2} X_{ij} \qquad j = 2, 3\ldots\ n+2 \qquad (5)$$

$$\sum_{i=1}^{n+2} X_{ij} = 1 \qquad j = 2, 3\ldots\ n+2 \qquad (6)$$

$$\sum_{i=1}^{n+2} X_{ji} = 1 \qquad j = 2, 3\ldots\ n+2 \qquad (7)$$

$$\sum_{i=1}^{n+2} X_{ii} = 0 \qquad (8)$$

$$X_{ij} + X_{ji} \leq 1 \qquad i, j = 2, 3\ldots\ n+2 \qquad (9)$$

$$Y_i = X_{1i} \qquad i = 2, 3\ \ldots\ n+1 \qquad (10)$$

$$Z_{ij} = X_{ij} \qquad i, j = 2, 3\ldots\ n+1 \qquad (11)$$

$$RD_i = X_{i1} \qquad i = 2, 3\ \ldots\ n+1 \qquad (12)$$

$$X_{ij} = \{0, 1\} \qquad i, j = 2, 3\ldots\ n+2 \qquad (13)$$

- Constraints (1) through (9) perform the identical functions as the first 9 constraints in the previous model.
- Constraint (10) ensures variable Y is equal to variable X when the path is from depot to pallet i. Y is needed to identify the time element of picking the first pallet in the objective function.
- Constraint (11) ensures variable Z is equal to variable X when the path goes from one pallet to pick a second pallet. Z is needed to identify all the time elements in the objective function related to picking the second pallet to be stacked on the first pallet.
- Constraint (12) ensures variable RD is equal to variable X when the path returns from pallet i to depot. RD is needed to identify the time element in the objective function related to the return trip from picked pallet to the depot.

3.3.2.1 Numerical examples

The numerical examples here parallel those previously discussed as illustrations of how this model can be used and the type of information it yields. The values given the input parameters were found in [58] and are consistent with our experience: 1) The forklift travel rate (speed) in a warehouse is 150 feet per minute (s =150), and 2) The time required to pick a pallet from the rack is 0.3 minute (t_p =0.3). We further assume the time required to stack one pallet on another is 0.3 minute (t_s=0.3).

The model was generated by placing the middle pallets near the depot and the other pallets incrementally further by 10 distance units each. For example, in 70 pallets example, pallets 36 and 37 are 50 distance units away from depot, while pallets 35 and 38 are 60 distance units further from depot and similarly pallets 11 and 62 are 300 distance units away from depot. See Appendix for distance and stackability appendices.

70 pallets: The model will be used to solve a seventy pallet problem. This problem has as stackabilty matrix that is 72x72 and it is assumed there are 70 possible stacking options assigned randomly. The equivalent stackability density for this matrix is 1.45% which is 70 divided by 4830; the maximum number of one's allowing all pallets to be stackable on each other except on itself.

Distance matrix was similarly generated based on having 2 middle pallets 35 and 36 closest to depot then all other pallets were located at increments of 10 from each other. The optimal solution is:

D-P1-P2-D-P3-P4-D-P5-P6-D-P7-P8-D-P9-P10-D-P11-P12-D-P13-P14-D-P15-P16-D-P17-P18-D-P19-P20-D-P21-P22-D-P23-P24-D-P25-P26-D-P27-P28-D-P29-P30-D-P31-

P32-D-P33-P34-D-P35-P36-D-P37-P38-D-P39-P40-D-P41-P42-D-P43-P44-D-P45-P46-D-P47-P48-D-P49-P50-D-P51-P52-D-P53-P54-D-P55-P56-D-P57-P58-D-P59-P60-D-P61-P62-D-P63-P64-D-P65-P66-D-P67-P68-D-P69-P70-D-Dummy-D.

This solution requires a total travel and processing time of 145.83 minutes. For comparison, the single pick alternative requires total travel time of 226.33 minutes; hence, stacking reduces the required time by 80.5 minutes which would likely be considered significant in practice.

50 more examples: The model has been used to solve problems containing 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 pallets, each was solved 5 times using five different variations of stackability options; total of 50 examples. Results are shown below in Figure 3.3. Each data point below is average of 5 replications, each is run with a different stackability matrix, where ones are in the matrix are scattered differently in a random manner but sum of ones in all 5 different stackability matrices are equal to number of pallets (problem size). Distance matrices were similarly generated based on having 2 middle pallets closest to depot then all other pallets were located at increments of 10 from each other. The computational effort required to find optimal solution increases with problem size as shown in Figure 3.3. This is expected due to the combinatorial nature of underlying problem. Graph grows exponentially.

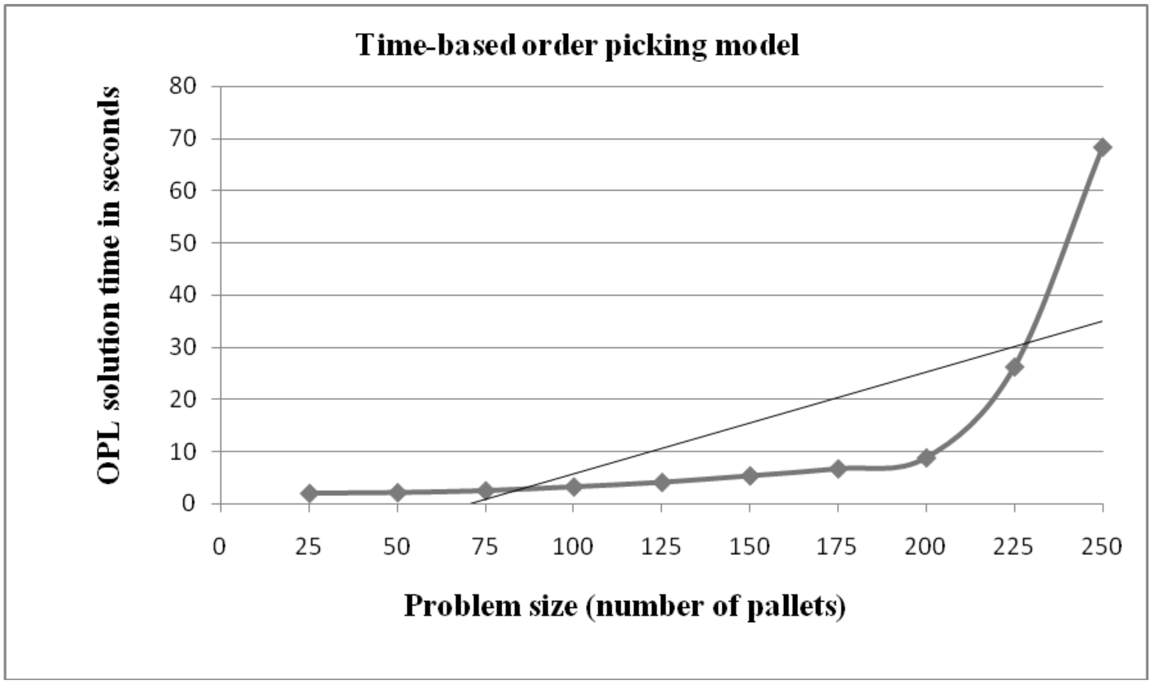


Figure 3.3: Time vs. size in picking time

3.3.3 Time-based storage model

A closely related problem to the previous one is when there are n pallets at the depot that need to be placed in the warehouse. From a modeling viewpoint, the storage and picking models are the same except for a few modifications. In particular, all that is required is to replace t_p and t_s by:

- $t_{l\,s}$ = time to store one pallet from a double stacked pallets setup on a rack in minutes
- t_l = time to store a single pallet on a rack in minutes

The decision variables have slightly different interpretations:

- X_{ij} = 1 if the path includes moving from node i to node j; 0 otherwise
- Y_i = 1 if pallet i is chosen to be stored; 0 otherwise

- Z_{ij} = 1 if pallet i is chosen to be stacked on top of another pallet; 0 otherwise
- RD_i = 1 if you return from pallet i to Depot; 0 otherwise

With these modifications, the time-based storage model objective function is:

Minimize

$$\sum_{k=2}^{n+1} c_{1k} * Y_k / s + \sum_{k=2}^{n+1} \mathrm{t}_l * Y_k + + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} \mathrm{t}_{l\,s} * Z_{kl} + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} c_{kl} * Z_{kl} / s + \sum_{k=2}^{n+1}\sum_{l=2}^{n+1} \mathrm{t}_l \; * Z_{kl} + \sum_{k=2}^{n+1} c_{k1} * RD_k / s$$

The constraint set is identical to the time-based model. This model was also presented in a conference proceeding [59].

3.3.3.1 Numerical Example for time-based storage model

These numerical examples use the following input parameters:

s =150, t_l =0.3 and $t_{l\,s}$ =0.5.

100 pallets: The model will be used to solve a problem with 100 pallets which has a 102x102 stackability matrix that contains 100 ones assigned randomly. The equivalent stackability density for this matrix is 1.01% which is 100 divided by 9900; the maximum number of one's allowing all pallets to be stackable on each other except on itself. Distance matrices were similarly generated based on having 2 middle pallets 50 and 51 closest to depot then all other pallets were located at increments of 10 from each other. The optimal storage path is:

D-P1-P2-D-P3-P4-D-P5-P6-D-P7-P8-D-P9-P10-D-P11-P12-D-P13-P14-D-P15-P16-D-P17-P18-D-P19-P20-D-P21-P22-D-P23-P24-D-P25-P26-D-P27-P28-D-P29-P30-D-P31-P32-D-P33-P34-D-P35-P36-D-P37-P38-D-P39-P40-D-P41-P42-D-P43-P44-D-P45-P46-D-P47-P48-**D-P49-D-P50-D-P51-D-P52-D**-P53-P54-P55-P56-D-P57-P58-D-P59-P60-

D-P61-P62-D-P63-P64-D-P65-P66-D-P67-P68-D-P69-P70-D-P71-P72-D-P73-P74-D-P75-P76-D-P77-P78-D-P79-P80-D-P81-P82-D-P83-P84-D-P85-P86-D-P87-P88-D-P89-P90-D-P91-P92-D-P93-P94-D-P95-P96-D-P97-P98-D-P99-P100-D-Dummy-D.

The section of the route that is in bold is of interest because these four pallets (49, 50, 51 and 52) were stored singly even though some could be stacked. The reason is that they were located sufficiently close to the Depot that the overall time is shorter to store them singly than stacking and incurring the additional time with the related tasks. The total time associated with this solution is 268.13 minutes that is considerably less than the 423.33 minutes that would be required to store each singly.

50 more examples: The model has been used to solve problems containing 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 pallets, each was solved 5 times using five different variations of stackability options; total of 50 examples. Results are shown below in Figure 3.4. Each data point below is average of 5 replications, each is run with a different stackability matrix, where ones are in the matrix are scattered differently in a random manner but sum of ones in all 5 different stackability matrices are equal to number of pallets (problem size). Distance matrices were similarly generated based on having 2 middle pallets closest to depot then all other pallets were located at increments of 10 from each other.

The computational effort required to find optimal solution increases with problem size as shown in Figure 3.4. This is expected due to the combinatorial nature of underlying problem. Graph grows exponentially.

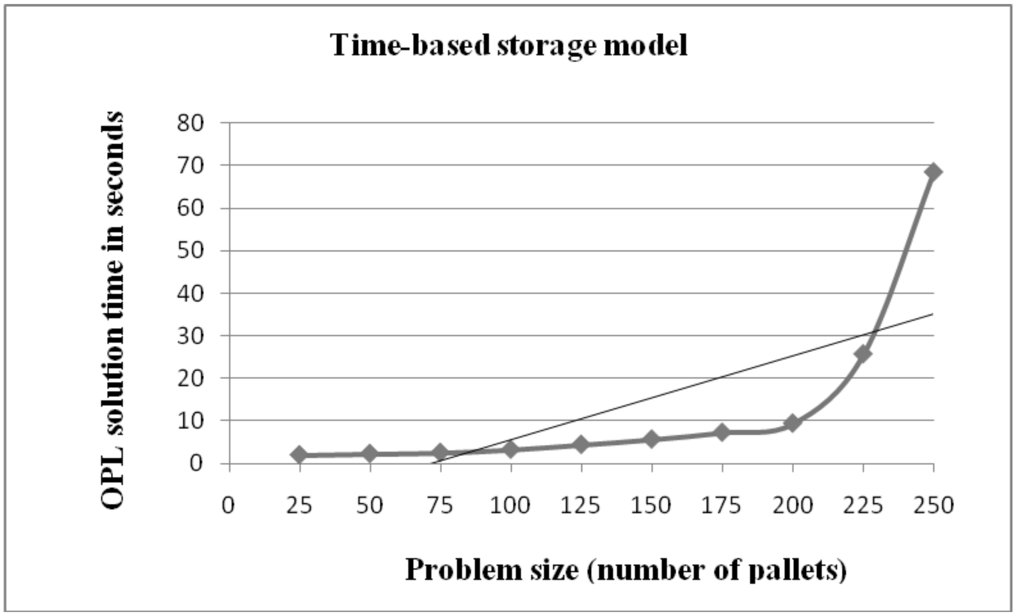


Figure 3.4: Time vs. size in storage time

3.4 Computational effort

The nonpolynomial increase in the required computing time to find an optimal solution as the number of pallets increases has been illustrated for each model. The underlying combinatorial nature of these problems certainly explains this result. Another aspect of computing time that is explored is the impact on time to find the optimal solution as the number of double stacking opportunities in the stackability matrix increases; that is, how many ones were included relative to the number of zeros. This idea is common in many areas including the flow dominance concept, which was originally introduced by Vollmann and Buffa [60].

A two hundred and fifty pallets example is used to investigate the impact of increasing stackability density on computation time shown in Figure 3.5. Thirty Five runs were made; five replications were run per each of the seven stackability densities 0.1, 0.5,1,5,10,15, and 20. Distance matrix was similarly generated based on having 2 middle pallets 125 and 126 closest to depot then all other pallets were located at increments of 10 from each other.

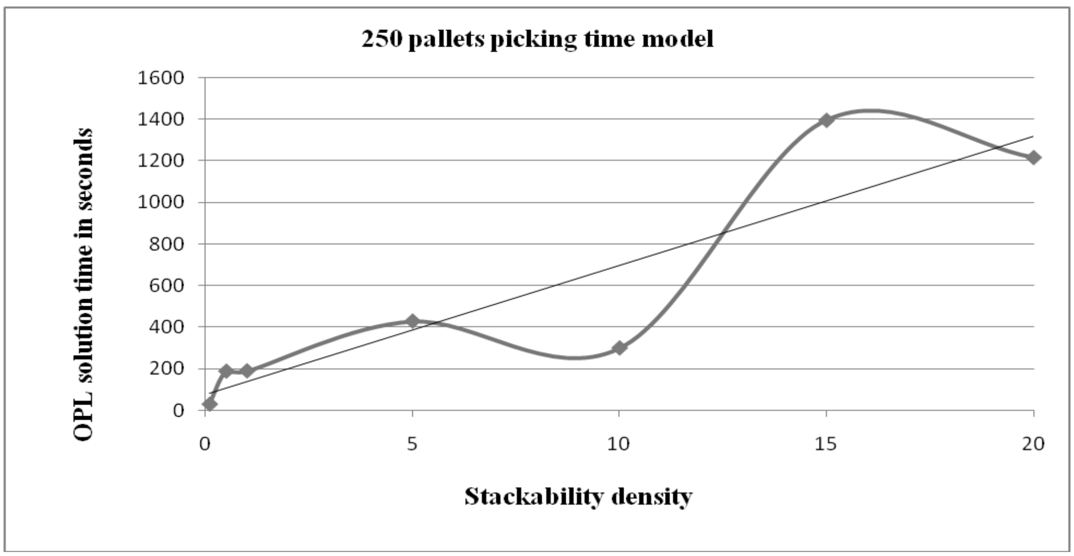


Figure 3.5: Time vs. stack -picking time

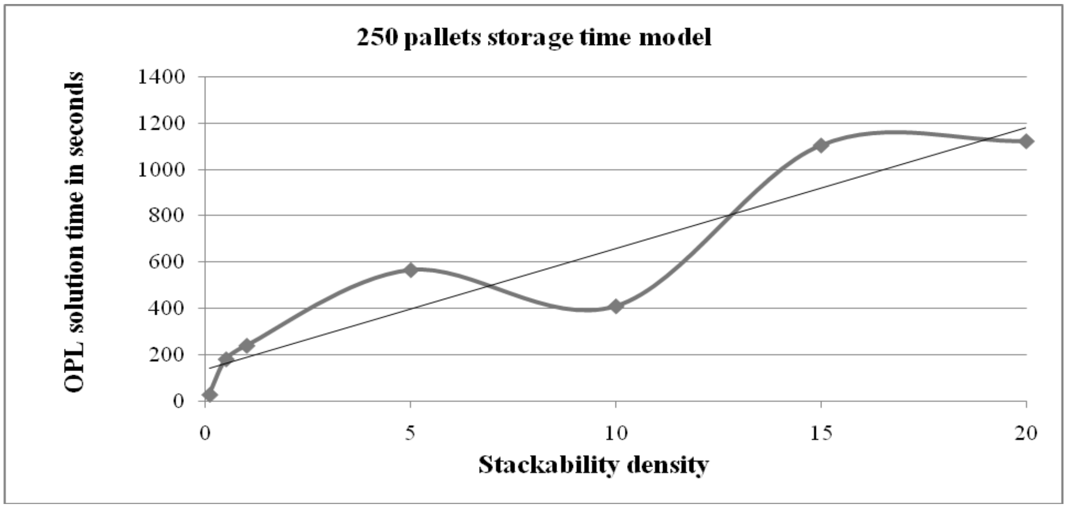


Figure 3.6: Time vs. stack -storage time

Each of the five replications had the same stackability density but ones were scattered differently in the matrix. Figures 3.5, 3.6 and 3.7 show a trend; as the stackability density increases the solution time increases.

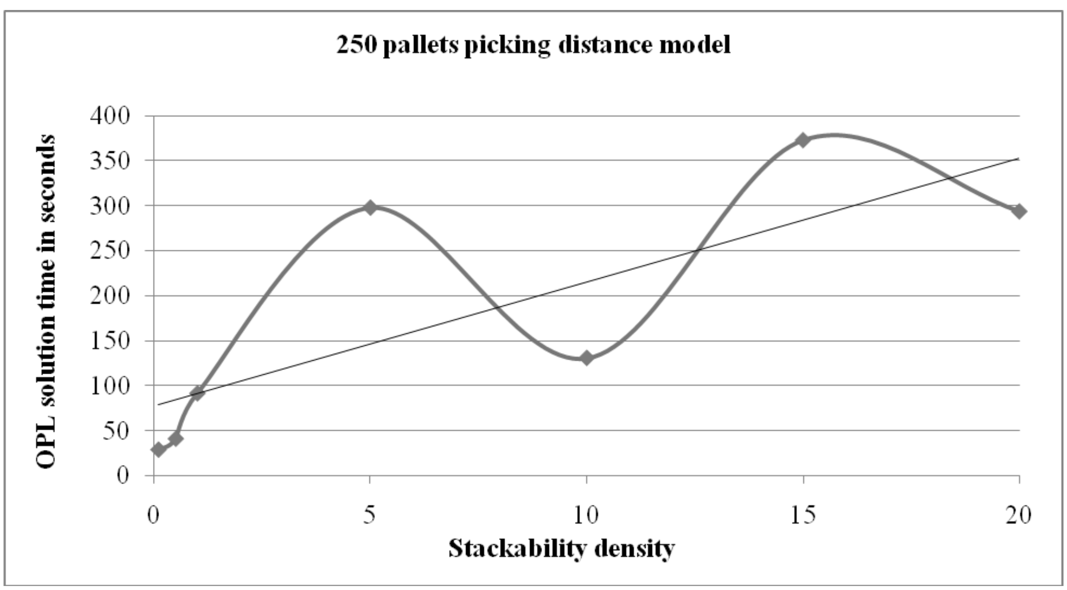


Figure 3.7: Time vs. stack -picking distance

This matches intuition as increasing stackability density, increases number of possible paths which increases the number of branches that have to be addressed in the OPL algorithm which increases the computational time.

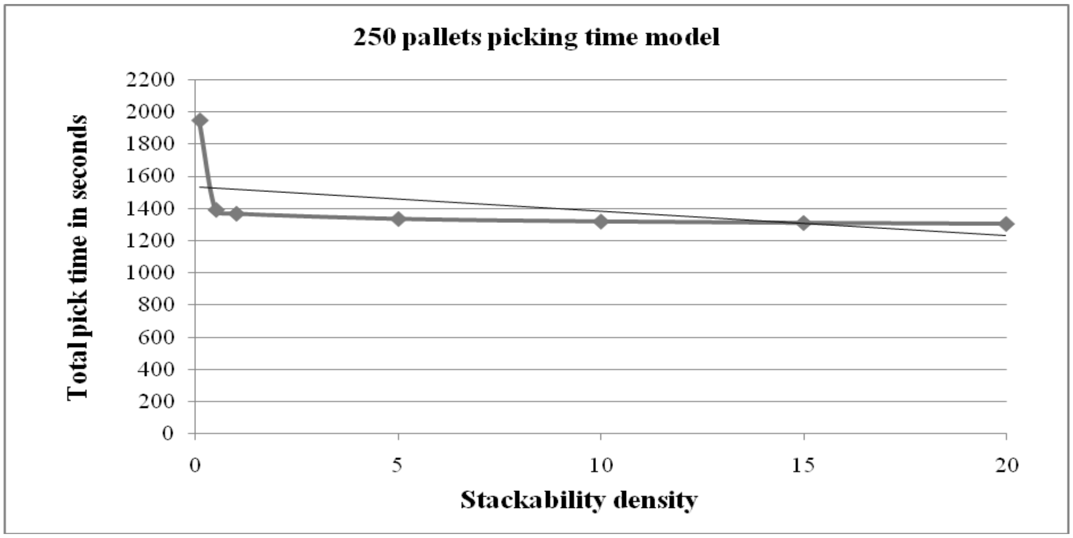


Figure 3.8: Stackability vs. total time

Figure 3.8 shows as the stackability density increases the total pick time decreases.

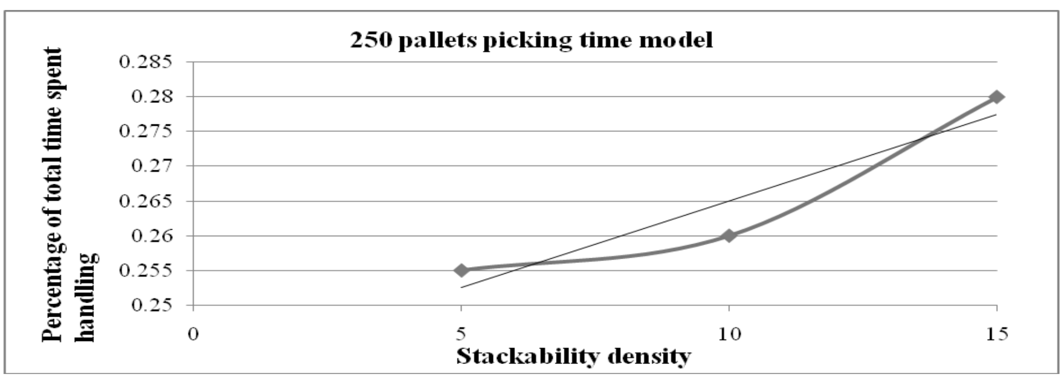


Figure 3.9: Stackability vs. handling percent

Figure 3.9 shows as the stackability density increases the Percentage of total time spent handling increases, which is logical as more pallets are double stacked i.e. more handling.

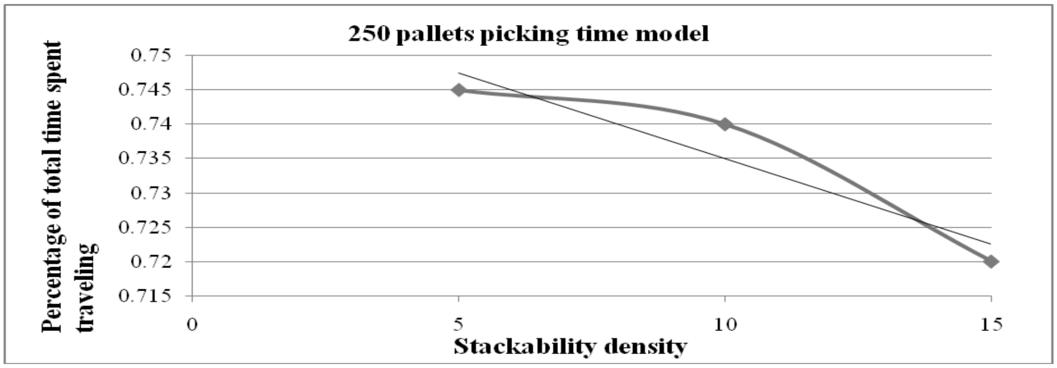


Figure 3.10: Stackability vs. travel percent

Figure 3.10 shows as the stackability density increases the percentage of total time spent traveling decreases, which is logical as more pallets are double stacked leading to less traveling as two single pick trips are combined into one trip.

3.5 Conclusions and Future Research

In this research three mathematical models were developed. One model analyzed the order picking operation in terms of travel distance. A second model analyzed the order picking operation in terms of time, adding the time elements of picking and stacking a pallet and a third model analyzed the order storage operation in terms of time. The concept of pallet stackabilty was applied in all three models. We conclude that these three mathematical models are working validated models, as they were coded in OPL software and solved for 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 pallet size problems successfully, each was solved 5 times using five different variations of stackability options; total of 150 examples for three models. Problems with a small number of pallets had the solution confirmed by complete enumeration.

In the future research combined order picking and storage model will be developed where forklift driver performs order picking and order storage of pallets simultaneously in the same trip in the model, called dual command in literature.

CHAPTER FOUR

COMBINED ORDER PICKING AND STORAGE WITH STACKABLE PALLETS

Abstract

In this chapter we are reducing the time spent in performing the order picking and storage activities in a warehouse. In real life in a warehouse, a forklift driver would pick and store full pallets alternatively in the same trip. In this paper we model the dual command process which is the combination of both picking and storage activities in the same trip. This process occurs in a manual warehouse where full pallets can be double stacked. The mathematical model allows the forklift driver to either start with a store or a pick move followed by more store or pick moves according to the shortest time route that the model will recommend. The model finds the optimal path that leads to the minimum travel time. Using this model with pallet stackability, daily material handling costs can be cut down almost in half compared to single pallet operations. These cost savings would improve the economy of the company that owns the warehouse.

Keywords

Dual command, order-picking, storage, path optimization, full pallet, time model

4.1 Introduction

In the model, a typical trip by a forklift driver would be leaving depot with one or two or no storage pallets, then first move would be either storing one pallet or picking a pallet then second move would be either storing a second pallet or picking another pallet, then remaining moves would alternate between storing and picking until at the maximum two pallets are stored and another two pallets are picked then finally last move is to return to the depot. The forklift can carry a maximum of two pallets at anytime, which can be two storage pallets, or one storage pallet and one picking pallet, or two picking pallets. The minimum travel time outcome of the model will help the warehouse management team reduce their expenses and improve their economic situation.

4.2 Literature Review

In literature, dual command is the keyword used to stand for combining storage and picking in one trip by the forklift driver. In dual command, workers travel loaded from the pickup and deposit (P&D) point first to a location to store a pallet, then to a second location from which they pick a pallet and return to the P&D point. We name the P&D point in our research depot. On the other hand, single command stands for performing either storage or picking in one trip by the forklift driver in the warehouse.

In 1983, Ratliff and Rosenthal [10] used graph theory as a solution approach to address the order picking problem, which is the problem of minimizing distance or time traveled by the material handling vehicle while picking ordered items stored in a warehouse then transporting them to the shipping area. Many papers came after that on order picking. For example, J.P. van den Berg and W.H.M. Zijm [8] introduced some of the order picking models and problem classifications. Also, in 1993, Hwang and Song [26] developed the expected travel time models based on the probabilistic analysis for single and dual commands assuming randomized storage assignment policy. In our research, we don't use probability when dealing with single and dual commands as we let the model choose whichever would lead to shorter picking time. Sometimes a forklift would take shorter time to single pick stackable pallets one at a time that are too close to the depot instead of double stacking them. While in another situation a forklift would take shorter time to pick double stacked pallets together if the pallets are far from the depot.

On the other hand, storage is the process of assigning items to their storage locations in a warehouse. Most of the models developed for storage are for automated warehousing which is not the topic of our research as we focus here on manual warehousing. However, there was a paper by Queirolo et al. [31] who developed a simulation model to solve the warehouse layout optimization problem, which determines where to assign storage areas in a warehouse to different classes of items to reduce travel time. The model they developed reduces the global storage cost through minimizing the total travel time. Queirolo et al. [31] addressed a different problem which is the warehouse layout optimization problem while our research addresses the storage optimization in a warehouse using stackable pallets.

In 2009, Pohl et al. [35] developed an expression for expected travel distance for dual command operations. Pohl et al. [35] used the expression for expected travel distance to analyze three common warehouse designs. Pohl et al. [35] concluded that warehouse design layout C in their paper is the best layout. Layout C had racks parallel to the shipping dock with aisles perpendicular to the shipping dock. Their conclusion confirmed our choice of this warehouse layout C for our problem, besides it was the same layout that I have seen at the Welcome road external warehouse at Robert Bosch Anderson plant in South Carolina.

On the other hand, Malmborg and Al-Tassan [61] developed an integrated model to study the impact of item, equipment, storage configuration and operating parameters in less than unit load order picking systems. They combined the travel time and storage space models to estimate order picking cycle times from which the impact of alternative interleaving disciplines can be evaluated. In our research we are combining the order picking and storage for a full pallet size order picking problem not for small boxes or partially filled pallets; less than a unit load problem. Besides they are looking into order picking and storage space not order storage like in our research. Also, Bozer and White [62] developed travel time models for automated storage/retrieval (AS/R) machines; however in our research we focus on manual warehouse systems not automated ones.

Two mathematical time models were developed for both the order-picking and storage activities separately in chapter three by Hassan Aly and Ferrell and the storage model was also presented by Hassan Aly in [59]. In this chapter we will combine these two models adding new constraints to allow the performance of both order-picking and order-storage activities alternatively in the same trip made by the forklift driver in the warehouse, dual command.

4.3 The Research Problem

This research focuses on determining the optimal picking and storage route for an order picker on a forklift in a manual warehouse that handles pallet loads. An important assumption is that at most two pallets can be stacked on top of each other and the storage locations can accommodate exactly one pallet. It is also assumed that pallets are available that must be stored and picked so performing both tasks in a single route is acceptable. Since the time required to manipulate the pallets can have a significant impact on the overall route time, a time based approach rather than a distance approach is required to determine the minimum total time to store and pick a set of pallets. As discussed earlier, a mixed integer programming model is developed that allows pallet storage and picking on the same route. "Stackability" is a concept we add to the model that allows double stacking two pallets together. For example, if a forklift driver is picking two stackable pallets, he can pick one pallet from a rack, carry it to the location of the other pallet, put the first pallet on the ground, pick the other pallet from its rack, stack the two pallets together, and then pick both pallets stacked on top of each other and move them to the depot area. An analogous scenario exists for storing two pallets. The research problem is to determine the pallets to stack and the ones to deliver unstacked as well as the route that achieves the shortest total time to pick and store a set of known orders.

4.4 Optimization Model

4.4.1 Introduction

In this section, a combined order picking and storage time model is presented. This model minimizes the total time consumed by the forklift driver performing both storage and order picking in a warehouse. When at most two pallets can be stacked on each other, there are a limited number of feasible paths that can be performed and the model includes all feasible paths which came out to be 12 paths. These paths are reflected in Figure 4.1. In this figure, it is assumed that all routes start and end at the depot while a store pallet is S and a pick pallet is P. The number simply identifies which pallet is present, for example, S1 is first pallet to be stored.

Leave depot with TWO pallets to be stored:

- S1 – S2
- S1 – S2 – P1
- S1 – P1 – S2
- S1 – S2 – P1 – P2
- S1 – P1 – S2 – P2

Leave depot with ONE pallet to be stored:

- S1
- S1 – P1
- P1 – S1
- S1 – P1 – P2
- P1 – S1 – P2

Leave depot with ZERO pallets to be stored:

- P1
- P1 – P2

Figure 4.1 Twelve feasible paths in the model

Figure 4.2 diagrams a few of the options so the reader can imagine the maneuvers required at each stop.

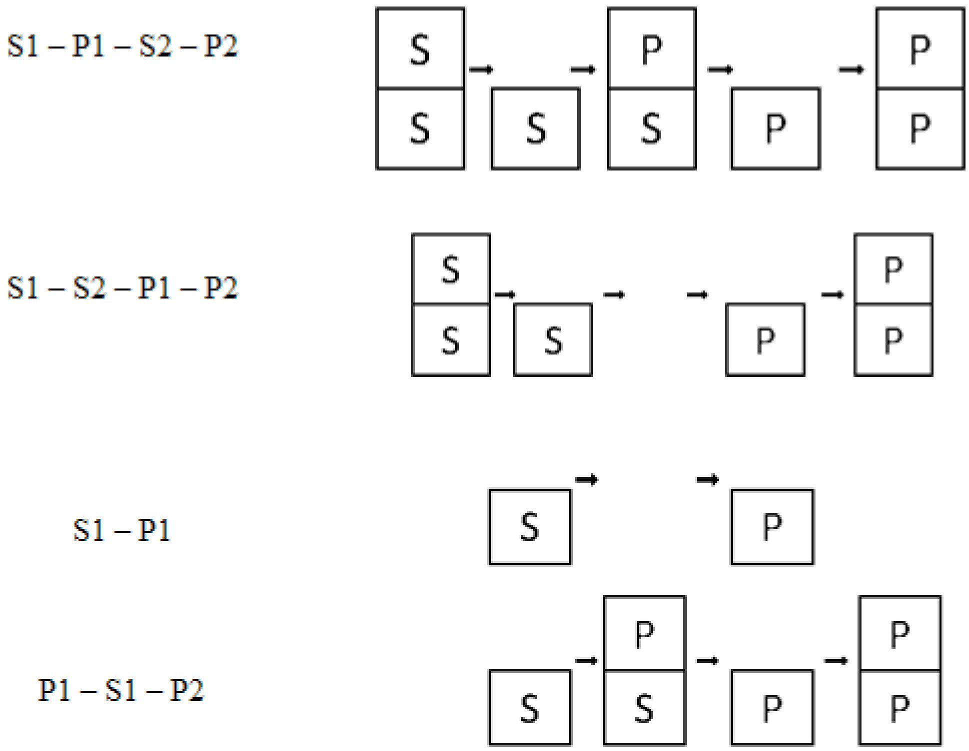


Figure 4.2 Diagram of few feasible paths

4.4.2 Assumptions

The model utilizes the following assumptions:

- One forklift is performing the order picking and storage functions and each route begins and ends at the depot.
- There is a one-to-one mapping between locations and pallets and each pallet has a unique location
- Pallets to be stored have their locations free and pallets to be picked are in different locations from pallets to be stored.
- All pallets to be stored are available at depot.

- Whether pallet i is stackable on pallet j or vice versa, the stacking can be accomplished at either location.
- When pallets are stacked to be stored, the pallet to be stored first is on the top
- A maximum of two pallets may be stacked.

4.4.3 Mathematical model

In this model, indexes identify locations in the warehouse as follows: 1) the depot is location 1, 2) the ns pallets are to be stored in locations 2 through ns+1 and 3) the np pallets to be picked are in locations ns+2 through ns+np+1. To accommodate the fact that we don't know exactly how many routes are required a priori, a dummy location is at ns+np+2. Now, these locations don't represent physical locations in the warehouse, only a location (anywhere in the warehouse but known) that a pallet needs to be stored or picked. This model required the following input parameters:

- Co = travel and material handling times associated with pick/storage options.
- d_{ij} = the distance between location i and location j in feets.
- S_{ij} = 1 if pallet i is stackable on j; 0 otherwise
- t_{ss} = time to store one pallet from a double stack setup on a rack in minutes
- t_s = time to store a single pallet on a rack in minutes
- s = average speed of the forklift traveling in a warehouse in feet per minute
- $t_{p\,=}$ time to pick a pallet in minutes
- $t_{ps\,=}$ time to pick a pallet and stack it on another pallet in minutes
- n = total number of pallets need to be stored and picked in a warehouse.

- ns = number of pallets to be stored
- np = number of pallets to be picked

The decision variables are all {0, 1} and define various parts of the route:

- X^{1}_{ij} = route is store i then store j.
- X^{2}_{ijk} = route is store i then store j then pick k.
- X^{3}_{ijk} = route is store i then pick j then pick k.
- X^{4}_{ijkl} = route is store i then store j then pick k then pick l.
- X^{5}_{ijkl} = route is store i then pick j then store k then pick l.
- X^{6}_{i} = route is store i
- X^{7}_{ij} = route is store i then pick j.
- X^{8}_{ij} = route is pick i then store j.
- X^{9}_{ijk} = route is store i then pick j then pick k.
- X^{10}_{ijk} = route is pick i then store j then pick k.
- X^{11}_{i} = route is pick i
- X^{12}_{ij} = route is pick i then pick j.
- All routes start and end at depot

C parameters used in model are:

$C_1 = (d_{1i} + d_{ij} + d_{j1})/s + t_{ss} + t_s$

$C_2 = (d_{1i} + d_{ij} + d_{jk} + d_{k1})/s + t_{ss} + t_s + t_p$

$C_3 = (d_{1i} + d_{ij} + d_{jk} + d_{k1})/s + 2t_{ss} + t_{ps}$

$C_4 = (d_{1i} + d_{ij} + d_{jk} + d_{kL} + d_{L1})/s + t_{ss} + t_s + t_p + t_{ps}$

$C_5 = (d_{1i} + d_{ij} + d_{jk} + d_{kL} + d_{L1})/s + 2t_{ss} + 2t_{ps}$

$C_6 = d_{1i}/s + t_s$

$C_7 = (d_{1i} + d_{ij} + d_{j1})/s + t_s + t_p$

$C_8 = (d_{1i} + d_{ij} + d_{j1})/s + t_{ss} + t_{ps}$

$C_9 = (d_{1i} + d_{ij} + d_{jk} + d_{k1})/s + t_s + t_p + t_{ps}$

$C_{10} = (d_{1i} + d_{ij} + d_{j1})/s + 2t_{ps}$

$C_{11} = (d_{1i} + d_{i1})/s + t_p$

$C_{12} = (d_{1i} + d_{ij} + d_{j1})/s + t_p + t_{ps}$

With these parameters defined, think of the objective function as containing sections of feasible routes and the constraints as ensuring the sections do not conflict. The model pieces together the segments in an optimal arrangement. The math model is as follows:

Minimize

$$Z = \sum_{i,j=2}^{ns+1} C_1 * \mathrm{X}^1{}_{\mathrm{ij}} + \sum_{i,j=2}^{ns+1} \sum_{k=ns+2}^{ns+np+1} C_2 * \mathrm{X}^2{}_{\mathrm{ijkl}} + \sum_{i,k=2}^{ns+1} \sum_{j=ns+2}^{ns+np+1} C_3 * \mathrm{X}^3{}_{\mathrm{ijk}} + \sum_{i,j=2}^{ns+1} \sum_{k,l=ns+2}^{ns+np+1} C_4 * \mathrm{X}^4{}_{\mathrm{ijkl}}$$

$$+ \sum_{i,j=2}^{ns+1} \sum_{k,l=ns+2}^{ns+np+1} C_5 * \mathrm{X}^5{}_{\mathrm{ijkl}} + \sum_{i=2}^{ns+1} C_6 * \mathrm{X}^6{}_{\mathrm{i}} + \sum_{i=2}^{ns+1} \sum_{j=ns+2}^{ns+np+1} C_7 * \mathrm{X}^7{}_{\mathrm{ij}} + \sum_{i=ns+2}^{ns+np+1} \sum_{j=2}^{ns+1} C_8 * \mathrm{X}^8{}_{\mathrm{ij}}$$

$$+ \sum_{i=2}^{ns+1} \sum_{j,k=ns+2}^{ns+np+1} C_9 * \mathrm{X}^9{}_{\mathrm{ijk}} + \sum_{i,k=ns+2}^{ns+np+1} \sum_{j=2}^{ns+1} C_{10} * \mathrm{X}^{10}{}_{\mathrm{ijk}} + \sum_{i=ns+2}^{ns+np+1} C_{11} * \mathrm{X}^{11}{}_{\mathrm{i}} + \sum_{i,j=ns+2}^{ns+np+1} C_{12} * \mathrm{X}^{12}{}_{\mathrm{ij}}$$

Subject to

$X_{ij} + X_{ji} \le S_{ij} + S_{ji}$, i =2, 3…ns+np+1, j=2, 3…ns+np+1 (1)

$$\sum_{i=1}^{ns+np+1} X_{ij} = 1\text{, j=2, 3…ns+np+1} \quad (2)$$

$$\sum_{j=1}^{ns+np+1} X_{ij} = 1\text{, i=2, 3…ns+np+1} \quad (3)$$

$$X_{ii} = 0\text{, i=1, 2…ns+np+2} \quad (4)$$

$$X_{ij} + X_{ji} \leq 1\text{, i =j=1, 2…ns+np+1} \quad (5)$$

$$X_{i1} = X_{1i}\text{, i =ns+np+2 (dummy)} \quad (6)$$

$$X_{ij} = 0{,}1\text{ , i=j = 2, 3… ns+np+2} \quad (7)$$

The constraints in this model perform the following functions:

1. Requires elimination of stacking that cannot occur if S is equal to 0 between any 2 pallets
2. Requires visiting each pallet once
3. Requires leaving each pallet once
4. Requires no pallet returns to itself or is stacked on itself (avoid cycling)
5. Avoids cycling between any 2 nodes except between dummy and depot to accommodate extra routes
6. Requires return to depot if dummy is used

4.4.4 Numerical Study

To illustrate some features of this model and provide some verification, we now provide several numerical examples. In all of these, ILOG OPL Development Studio version 5.5 was used on a Dell personal computer with an Intel Core 2 Duo processor and

2.00 GB of RAM. Numerical values for the inputs are the same as in [58] because they were also consistent with our experience: 1) The forklift travel rate (speed) in a warehouse is 150 feet per minute ($s = 150$), and 2) The time required to pick a pallet from the rack is 0.3 minute ($t_p = 0.3$). We further assume the time required to store a pallet on the rack is 0.3 minute ($t_s = 0.3$), the time to pick a pallet and stack it on another pallet is 0.5 minute ($t_{ps} = 0.5$) and the time to store a pallet on rack from a double stack is 0.5 minute ($t_{ss} = 0.5$).

5 pallets Example: The model was used to solve a five pallets problem, where pallets are located close to each other in proximity. This problem has 5 possible stacking options (assigned randomly) allowing all pallets to be stackable on each other, meaning a stackability density of 25%. The storage pallets are pallets number 1, 2 while the pick pallets are pallets number 3, 4, and 5. Distance matrix was generated based on layout. Each box in layout represents an increment of 5 feet distance. Please see appendix for layout and distance and stackability matrices. The solution presented below is an OPL solution generated after running for few seconds, it is a good solution as it beats both heuristics but no guarantee it's optimal. D represents the depot and i represents pallet i. Path is: D-1-2-5-D-4-3-D. This path can be translated as shown below. S means storage and P means picking. Path is: D-S-S-P-D-P-P-D which means:

1. Leave depot with pallets 1 and 2 - Store pallet 1 – Store pallet 2 - Pick pallet 5 – return to depot
2. Leave depot - Pick pallet 4 – Pick pallet 3 – return to depot

This solution requires a total travel and handling time of 4.1 minutes. For comparison, the travel time was 2.2 minutes; while the handling time was 1.9 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 46 % of the total time, which would likely be considered significant in practice when you add up all the different orders handled by a forklift driver which are at least hundreds of pallets daily.

Close pallets warehouse

7 pallets Example: The model was used to solve a seven pallets problem, where pallets are located close to each other in proximity. This problem has 7 possible stacking options (assigned randomly) allowing all pallets to be stackable on each other, meaning a stackability density of 16.67%. The storage pallets are pallets number 1, 2, 3, while the pick pallets are pallets number 4, 5, 6, and 7. Distance matrix was generated based on layout. Each box in layout represents an increment of 5 feet distance. Please see appendix for layout and distance and stackability matrices. The solution presented below is an OPL solution generated after running for 10.5 seconds, it is a good solution as it beats one heuristic and equals the other heuristic but no guarantee it's optimal. D represents the depot and i represents pallet i. Path is: D-2-5-D-3-1-7-D-4-6-D. This path can be translated as shown below. S means storage and P means picking. Path is: D-S-P-D-S-S-P-D-P-P-D which means:

1. Leave depot with pallet 2 - Store pallet 2 – Pick pallet 5 – return to depot

2. Leave depot with pallets 3 and 1 - Store pallet 3 – Store pallet 1 - Pick pallet 7 – return to depot
3. Leave depot - Pick pallet 4 – Pick pallet 6 – return to depot

This solution requires a total travel and handling time of 5.433 minutes. For comparison, the travel time was 2.93 minutes; while the handling time was 2.5 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 46 % of the total time, which would likely be considered significant in practice when you add up all the different orders handled by a forklift driver which are at least hundreds of pallets daily.

9 pallets Example: The model was used to solve a nine pallets problem, also close in location proximity. This problem has 9 possible stacking options (assigned randomly) allowing all pallets to be stackable on each other, meaning a stackability density of 12.5%. The storage pallets are pallets number 1, 2, 3, and 4 while the pick pallets are pallets number 5, 6, 7, 8 and 9. Distance matrix was generated based on layout. Each box in layout represents an increment of 5 feet distance. Please see appendix for layout and distance and stackability matrices. The solution presented below is an OPL solution generated after running only 386 seconds, it is a good solution as it beats both heuristics but no guarantee it's optimal. D represents the depot and i represents pallet i. Path is: D-1-7-9-D-2-5-D-3-8-D-4-6-D. This path can be translated as shown below. S means storage and P means picking. Path is: D-S-P-P-D-S-P-D-S-P-D-S-P-D which means:

1. Leave depot with pallet 1 - Store pallet 1 – Pick pallet 7 – Pick pallet 9 - return to depot
2. Leave depot with pallet 2 - Store pallet 2 - Pick pallet 5 – return to depot
3. Leave depot with pallet 3 - Store pallet 3 - Pick pallet 8 – return to depot
4. Leave depot with pallet 4 - Store pallet 4 – Pick pallet 6 – return to depot

This solution requires a total time of both traveling and handling of 6.7 minutes. For comparison, the travel time was 3.8 minutes; while the handling time was 2.9 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 43 % of the total time, which would likely be considered significant in practice when you add up all the different orders handled by a forklift driver which are at least hundreds of pallets daily.

Scattered pallets warehouse

8 pallets Example: The model was used to solve an eight pallets problem, where pallets are distant in location from each other. This problem has 56 possible stacking options (assigned randomly) allowing all pallets to be stackable on each other, meaning a stackability density of 100%. The storage pallets are pallets number 1, 2, 3, and 4, while the pick pallets are pallets number 5, 6, 7, and 8.

Distance matrix was generated based on layout. Each box in layout represents an increment of 5 feet distance. Please see appendix for layout and distance and stackability matrices. The solution presented below is an OPL solution generated after a very long run (over 88 hours), it is a good solution as it beats 2 heuristics but no guarantee it's optimal. D represents the depot and i represents pallet i. Path is: D-2-6-D-3-8-D-4-7-D-5-1-D. This path can be translated as shown below. S means storage and P means picking. Path is: D-S-P-D-S-P-D-S-P-D-P-S-D which means:

3. Leave depot with pallet 2 - Store pallet 2 – Pick pallet 6 – return to depot
4. Leave depot with pallet 3 - Store pallet 3 – Pick pallet 8 – return to depot
5. Leave depot with pallet 4 - Store pallet 4 – Pick pallet 7 – return to depot
6. Leave depot with pallet 1 - Pick pallet 5 – Store pallet 1 – return to depot

As you can see there are two pick/store options in this solution, option 5 (D-S-P-D) and option 6 (D-P-S-D). This solution requires a total travel and handling time of 12.6 minutes. For comparison, the travel time was 9.8 minutes; while the handling time was 2.8 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 22% of the total time, which would likely be considered significant in practice when you add up all the different orders handled by a forklift driver which are at least hundreds of pallets daily.

4.4.5 Increasing number of double stacking opportunities

The impact if any that increasing the stackability density has on the time required to find the optimal solution is now explored. This idea is common in many areas including the flow dominance concept, which was originally introduced by Vollmann and Buffa [60].

An example with 8 pallets close in distance is used to investigate the impact of increasing the number of double stacking opportunities on computation time and Figure 4.3 below reports the results. Each instance is based on average of three replications, and each replication has a stackability matrix with same stackability density but ones scattered differently in the three replications. The graph below shows that the increase of the stackability density increases the computational time. This is probably because as the number of ones in the stackability matrix increases this increases the possibilities for stacking and hence more search to be done by OPL to get optimal solution.

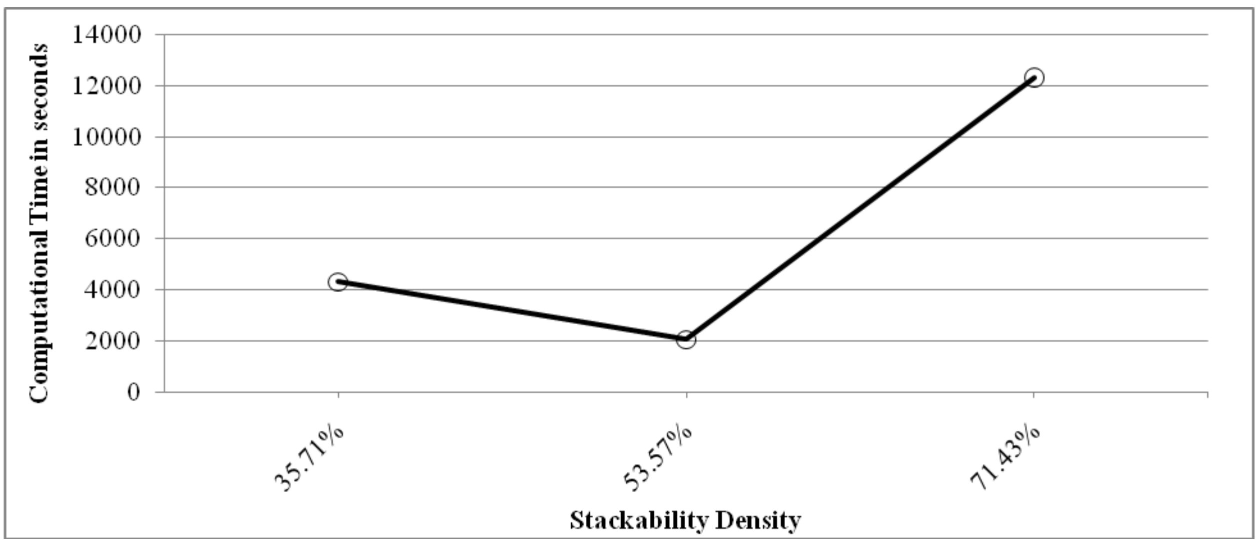


Figure 4.3: 8 close by pallets problem

4.5 Heuristics

Since most real world problems will involve at least 40 pallets (i.e., the number of double stacked pallets that are contained in a truck-load shipment) and likely many more, solving direct use of the optimization model is impossible. In this section, we describe two heuristics that have been developed in an effort to find good solutions with much less computation burden. The first heuristic simply executes the store and pick operations separately whereas the second heuristic allows the forklift driver to execute storage operations and picking operations in the same route although they cannot be alternated (i.e., store-pick-store is not allowed).

4.5.1 Heuristic 1 – Separate store and pick

Heuristic 1 is simply the time-optimal route for storing stackable pallets combined with the time-optimal route for picking. The mathematical programming models used to determine these routes are fully explained in chapter three. Intuitively, if a warehouse had a layout such that all the storage from a set of dock doors were on one side and the picks on the other – say raw materials on one side and finished goods on the other – then this heuristic and heuristic 2 as well will likely work really well if this type of segregation is seen but it is not an assumption of the model. When pallets to be stored and picked are mixed throughout the warehouse, it is hard to predict how effective this heuristic would be. Regardless, it is used for comparison both because it does have the potential to produce good results and because it solves to optimality with OPL within seconds for 8 pallets.

4.5.1.1 Heuristic 1 model

Although this heuristic is not a new model and it concatenates two models developed previously that find the time-optimal route for pure storage and pure picking, we use it to get quick answer to compare against combined model. Same was done for heuristic 2. Details of the individual models are provided in chapter three. As in the integrated model, pallets are numbered coincident with their locations with the first location being the depot, the next ns locations the pallets to be stored, the next np locations the pallets to be picked and the final location being the dummy used for modeling purposes. This model required the following input parameters:

- d_{ij} = the distance between location i and location j
- S_{ij} = 1 if pallet i is stackable on j; 0 otherwise
- t_{ss} = time to store one pallet from a double stack setup on a rack in minutes
- t_s = time to store a single pallet on a rack in minutes
- s = average speed of the forklift traveling in a warehouse in feet per minute
- $t_{p\,=}$ time to pick a pallet in minutes
- $t_{ps\,=}$ time to pick a pallet and stack it on another pallet in minutes
- ns = number of pallets to be stored
- np = number of pallets to be picked

The decision variables are:

- X_{ij} = 1 if the route includes pallet i to pallet j; 0 otherwise.
- Y_i = 1 if the route includes depot to pallet i; 0 otherwise.
- Z_{ij} = 1 the route includes pallet i to pallet j; 0 otherwise.
- RD_i = 1 if the route includes pallet i to depot; 0 otherwise.

Using these definitions, the following mathematical model will determine the optimal routes.

Minimize

$$Z = \sum_{i=2}^{ns+1} d_{1i} * Y_i / s + \sum_{i=2}^{ns+1} t_s * Y_i + \sum_{i,j=2}^{ns+1} d_{ij} * Z_{ij} / s + \sum_{i,j=2}^{ns+1} t_{ss} * Z_{ij} + \sum_{j=2}^{ns+1} d_{j1} * RD_j / s$$

$$+ \sum_{i=ns+2}^{ns+np+1} d_{1i} * Y_i / s + \sum_{i=ns+2}^{ns+np+1} t_p * Y_i + \sum_{i,j=ns+2}^{ns+np+1} d_{ij} * Z_{ij} / s + \sum_{i,j=ns+2}^{ns+np+1} t_{ps} * Z_{ij} + \sum_{j=ns+2}^{ns+np+1} d_{j1} * RD_j / s$$

Subject to

$$\sum_{i=2}^{ns+np+2} X_{1i} \geq 1 \qquad (1)$$

$$X_{ij} + X_{ji} \leq S_{ij} + S_{ji}, \text{ i, j =2, 3…ns+np+2} \qquad (2)$$

$$\sum_{i=1}^{ns+np+2} X_{ji} \geq X_{1j}, \text{ i =1, 2…ns+np+2, j =2, 3…ns+np+2} \qquad (3)$$

$$X_{(ns+np+2)1} = X_{1(ns+np+2)} \qquad (4)$$

$$X_{j1} \geq \sum_{i=2}^{ns+np+2} X_{ij}, \text{ j = 2, 3…ns+np+2} \qquad (5)$$

$$\sum_{i=1}^{ns+np+2} X_{ij} = 1, \text{ j =2, 3… ns+np+2} \qquad (6)$$

$$\sum_{j=1}^{ns+np+2} X_{ij} = 1, \text{ i =2, 3… ns+np+2} \qquad (7)$$

$$X_{ii} = 0, \text{ i =1, 2…ns+np+2} \qquad (8)$$

$$Y_i = X_{1i}, \text{ i = 2, 3…ns+np+1} \qquad (9)$$

$$Z_{ij} = X_{ij}, \text{ i, j =2, 3…ns+np+1} \qquad (10)$$

$$RD_i = X_{i1}, \text{ i = 2, 3…ns+np+1} \tag{11}$$

$$X_{ij} + X_{ji} = 0, \text{ i = 2, 3 …ns+1, j = ns+2, ns+3…ns+np+1} \tag{12}$$

$$X_{ij} = 0,1 \text{ , i, j = 1, 2, 3… ns+np+2, j = 2, 3… ns+np+2} \tag{13}$$

A short explanation regarding the purpose of the 13 constraints in this model are:

1. Starts enough routes so that each pallet can be singly picked or stored
2. Prevents stacking of pallets that are forbidden by the stackability matrix
3. Requires that the second move is either to a pallet or back to depot
4. Completes a no-cost route to the dummy if all pallets have been moved
5. Requires return to depot after only one stacking arrangement or no stacking is made for each pallet
6. Requires picking each pallet once
7. Requires leaving each pallet once
8. Requires no pallet is stacked on itself
9. Requires variable Y is equal to variable X when a the forklift moves from depot to pick a pallet i
10. Requires variable Z is equal to variable X when a the forklift moves from first stored or picked pallet to store or pick a second pallet i
11. Requires variable RD is equal to variable X when a the forklift returns from pallet i to depot
12. Prevents any move between locations of storage and pick pallets

4.5.2 Heuristic 2 – Limited dual commands

The second heuristic allows a single route to include both storage and picking activities; however, they cannot alternate. That is, a forklift driver can leave the depot with 2 stacked pallets to be stored, store them, pick two pallets (stacking them at the second location) and returning to the depot. This can be modeled by again concatenating the pure pick and pure store models but this time adding a mid-zone location that the route must include where the mode is changed from store to pick.

4.5.2.1 Heuristic 2 model

The locations for the pallets and the parameters for this model are identical to Heuristic 1 – see section 4.5.1.1 for details. The decision variables are slightly different:

- X_{ij} = 1 if you go from pallet i to pallet j; 0 otherwise.
- ZL_{ij} = 1 if you go from storage pallet i to storage pallet j; 0 otherwise.
- ZP_{ij} = 1 if you go from pick pallet i to pick pallet j; 0 otherwise.

The model for this heuristic is as follows:

Minimize

$$Z = \sum_{i=2}^{ns+1} d_{1i} * X_{1i} / s + \sum_{i=2}^{ns+1} t_s * X_{1i} + \sum_{i,j=2}^{ns+1} d_{ij} * ZL_{ij} / s + \sum_{i,j=2}^{ns+1} t_{ss} * ZL_{ij} + \sum_{j=2}^{ns+1} d_{j_{(ns+2)}} * X_{j(ns+2)} / s$$

$$+ \sum_{k=ns+3}^{ns+np+2} d_{(ns+2)k} * X_{(ns+2)k} / s + \sum_{k=ns+3}^{ns+np+2} t_p * X_{(ns+2)k} + \sum_{k,m=ns+3}^{ns+np+2} d_{km} * ZP_{km} / s + \sum_{k,m=ns+3}^{ns+np+2} t_{ps} * ZP_{km}$$

$$+\sum_{m=ns+3}^{ns+np+2} d_{m1} * X_{m1} / s$$

Subject to

$$\sum_{i=2}^{ns+1} X_{1i} \geq 1 \quad (1)$$

$$X_{ij} + X_{ji} \leq S_{ij} + S_{ji}\text{, i, j =2, 3…ns+1} \quad (2)$$

$$X_{km} + X_{mk} \leq S_{km} + S_{mk}\text{, k, m =ns+3, ns+4…ns+np+2} \quad (3)$$

$$\sum_{j=2}^{ns+1} X_{ij} + X_{i(ns+2)} \geq X_{1i}\text{, i =2, 3…ns+1} \quad (4)$$

$$\sum_{m=ns+3}^{ns+np+2} X_{km} + X_{k1} \geq X_{(ns+2)k}\text{, k = ns+3, ns+4…ns+np+2} \quad (5)$$

$$X_{j(ns+2)} \geq \sum_{i=2}^{ns+1} X_{ij}\text{, j =2, 3…ns+1} \quad (6)$$

$$\sum_{k=ns+3}^{ns+np+2} X_{(ns+2)k} = \sum_{j=2}^{ns+1} X_{j(ns+2)} \quad (7)$$

$$\sum_{i=2}^{ns+1} X_{ij} + X_{1j} = 1\text{, j =2, 3…ns+1} \quad (8)$$

$$\sum_{k=ns+3}^{ns+np+2} X_{km} + X_{(ns+2)m} = 1\text{, m = ns+3, ns+4…ns+np+2} \quad (9)$$

$$\sum_{j=2}^{ns+1} X_{ij} + X_{i(ns+2)} = 1\text{, i =2, 3…ns+1} \quad (10)$$

$$\sum_{m=ns+3}^{ns+np+2} X_{km} + X_{k1} = 1\text{, k = ns+3, ns+4…ns+np+2} \quad (11)$$

$X_{ii} = 0$, i = 2, 3 …ns+np+2 (12)

$ZL_{ij} = X_{ij}$, i, j = 2, 3 …ns+1 (13)

$ZP_{ij} = X_{ij}$, i, j = ns+3, ns+4…ns+np+2 (14)

$X_{1i} + X_{ij} + X_{j(ns+2)} \leq 3$, i, j = 2, 3 …ns+1 (15)

$X_{(ns+2)k} + X_{km} + X_{m1} \leq 3$, k, m= ns+3, ns+4…ns+np+2 (16)

$$\sum_{i=ns+3}^{ns+np+2} X_{i1} \geq 1 \quad (17)$$

$X_{ij} + X_{ji} \leq 1$, i, j = 2, 3 … ns+np+2 (18)

$X_{m1} \geq \sum_{k=ns+3}^{ns+np+2} X_{km}$, m = ns+3, ns+4…ns+np+2 (19)

$X_{ij} =$ 0,1 , i, j = 1, 2, 3… ns+np+2, j = 2, 3… ns+np+2 (20)

The constraints in this model perform the following functions:

1. Requires a first move to store a pallet
2. Constraints (2) and (3) require elimination of stacking that cannot occur if S is equal to 0
4. Requires a second move after storing one pallet
5. Requires a second move to pick a second pallet
6. Requires going to dummy after storing pallets
7. Requires going from dummy to pick pallets
8. Constraints (8) and (9) require visiting each pallet once
10. Constraints (10) and (11) require leaving each pallet once

12. Requires no pallet returns to itself or is stacked on itself (Avoids cycling)
13. Requires variable ZL is equal to variable X when forklift moves from first stored pallet to store a second pallet
14. Requires variable ZP is equal to variable X when forklift moves from first picked pallet to pick a second pallet
15. Requires return to depot after storing 2 pallets
16. Requires return to depot after picking 2 pallets
17. Requires a return move from pick pallet to depot
18. Prevents cycling between any 2 nodes (pallet or dummy)
19. Requires going from picking pallets to depot

4.6. Numerical Examples:

Heuristic 1: 5 pallets: The model of heuristic 1 will be used to solve a problem with 5 pallets which has a stackability matrix that contains 5 ones, meaning a stackability density of 25%. The storage pallets are pallets number 1 and 2 while the pick pallets are pallets number 3, 4, and 5. The optimal path is presented below. D represents the depot and i represents pallet i.

D-1-2-D-4-3-D-5-D

This path can be translated as shown below. S means storage and P means picking.

D-S-S-D-P-P-D-P-D

This solution requires a total travel plus handling time of 4.9 minutes. For comparison, the travel time was 3 minutes; while the handling time was 1.9 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 38.7 % of the total time, which would likely be considered significant in practice. This is good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Heuristic 2: 5 pallets: The model of heuristic 2 will be used to solve the same problem with 5 pallets. The optimal path is presented below. D represents the depot, Du represents dummy (middle zone) and i represents pallet i.

D-1-Du-3-D-2-Du-4-5-D

This path can be translated as shown below. S means storage and P means picking.

D-S-Du-P-D-S-Du-P-P-D

This solution requires a total travel and handling time of 4.2 minutes. For comparison, the travel time was 2.5 minutes; while the handling time was 1.7 minutes. The handling time is almost 40 % of the total time, which would likely be considered significant in practice. This is again good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Summary

Table below show results of 5 pallets problem

5 pallets problem	Total time in minutes
Combined Model	4.1
Heuristic 1	4.9
Heuristic 2	4.2

Table 4.1: Results in 5 pallets problem

Heuristic 1: 7 pallets: The model of heuristic 1 will be used to solve a problem with 7 pallets which has a stackability matrix that contains 7 ones, meaning a stackability density of 16.67%. The storage pallets are pallets number 1, 2, and 3 while the pick pallets are pallets number 4, 5, 6, and 7. The optimal path is presented below. D represents the depot and i represents pallet i.

D-1-3-D-2-D-4-6-D-5-D-7-D

This path can be translated as shown below. S means storage and P means picking.

D-S-S-D-S-D-P-P-D-P-D-P-D

This solution requires a total travel plus handling time of 6.833 minutes. For comparison, the travel time was 4.33 minutes; while the handling time was 2.5 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 36 % of the total time, which would likely be considered significant in practice. This is good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Heuristic 2: 7 pallets: The model of heuristic 2 will be used to solve the same problem with 7 pallets. The optimal path is presented below. D represents the depot, Du represents dummy (middle zone) and i represents pallet i.

D-1-Du-4-6-D-2-Du-5-D-3-Du-7-D

This path can be translated as shown below. S means storage and P means picking.

D-S-Du-P-P-D-S-Du-P-D-S-Du-P-D

This solution requires a total travel and handling time of 5.43 minutes. For comparison, the travel time was 3.13 minutes; while the handling time was 2.3 minutes. The handling time is almost 42 % of the total time, which would likely be considered significant in practice. This is again good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Summary

Table below show results of 7 pallets problem for combined model with 2 heuristics

7 pallets problem	Total time in minutes
Combined Model	5.4333
Heuristic 1	6.8333
Heuristic 2	5.4333

Table 4.2: Results in 7 pallets problem

Heuristic 1: 8 scattered pallets: The model of heuristic 1 will be used to solve a problem with 8 pallets which has a 10x10 stackability matrix that contains 56 ones, meaning a stackability density of 14.29%. The storage pallets are pallets number 1, 2, 3, and 4 while the pick pallets are pallets number 5, 6, 7, and 8. The optimal path is presented below. D represents the depot and i represents pallet i.

D-1-3-D-2-4-D-7-6-D-8-5-D

This path can be translated as shown below. S means storage and P means picking.

D-S-S-D-S-S-D-P-P-D-P-P-D

This solution requires a total travel and handling time of 14.6 minutes. For comparison, the travel time was 11.4 minutes; while the handling time was 3.2 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 22 % of the total time, which would likely be considered significant in practice. This is good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Heuristic 2: 8 scattered pallets: The model of heuristic 2 will be used to solve the same problem with 8 pallets. The optimal path is presented below. D represents the depot, Du represents dummy (middle zone) and i represents pallet i.

D-1-3-Du-6-7-D-2-4-Du-8-5-D

This path can be translated as shown below. S means storage and P means picking.

D-S-S-Du-P-P-D-S-S-Du-P-P-D

This solution requires a total travel and handling time of 13.4 minutes. For comparison, the travel time was 10.2 minutes; while the handling time was 3.2 minutes. The handling time is almost 24 % of the total time, which would likely be considered significant in practice. This is again good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Summary

Table below show results of 8 scattered pallets problem for combined model with 2 heuristics

8 pallets problem	Total time in minutes
Combined Model	12.6
Heuristic 1	14.6
Heuristic 2	13.4

Table 4.3: Results in 8 pallets problem

The results show that combined model beats heuristics 1 and 2. Also you can see figure 4.4 below showing combined model solution progress during running time for 8 scattered pallets.

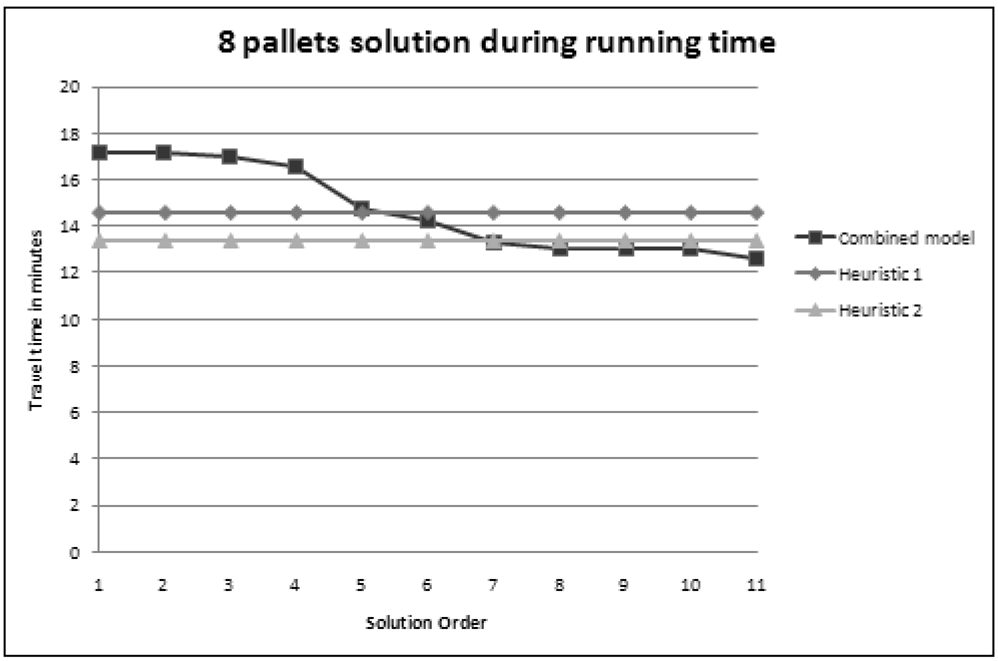


Figure 4.4: Solution during running time

Heuristic 1: 9 pallets: The model of heuristic 1 will be used to solve a problem with 9 pallets which has a stackability matrix that contains 9 ones, meaning a stackability density of 12.5%. The storage pallets are pallets number 1, 2, 3, and 4 while the pick pallets are pallets number 5, 6, 7, 8 and 9. The optimal path is presented below. D represents depot and i represents pallet i.

D-1-3-D-2-D-4-D-5-D-6-D-7-9-D-8-D.

This path can be translated as shown below. S means storage and P means picking.

D-S-S-D-S-D-S-D-P-D-P-D-P-P-D-P-D

This solution requires a total travel and handling time of 8.967 minutes. For comparison, the travel time was 5.87 minutes; while the handling time was 3.1 minutes. This model considered the handling time which was not considered in previous models. The handling time is around 34 % of the total time, which would likely be considered significant in practice. This is good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Heuristic 2: 9 pallets: The model of heuristic 2 will be used to solve the same problem with 9 pallets. The optimal path is presented below. D represents the depot, Du represents dummy (middle zone) and i represents pallet i.

D-1-Du-5-D-2-Du-6-D-3-Du-7-9-D-4-Du-8-D

This path can be translated as shown below. S means storage and P means picking.

D-S-Du-P-D-S-Du-P-D-S-Du-P-P-D-S-Du-P-D

This solution requires a total time both traveling and handling of 6.9 minutes. For comparison, the travel time was 4 minutes; while the handling time was 2.9 minutes. The handling time is almost 42 % of the total time, which would likely be considered significant in practice. This is again good as it emphasizes the importance of this research modeling the handling time which was not done before in literature.

Summary

Table below show results of 9 pallets problem for combined model with 2 heuristics

9 pallets problem	Total time in minutes
Combined Model	6.7
Heuristic 1	8.9667
Heuristic 2	6.9

Table 4.4: Results in 9 pallets problem

4.7 Conclusions and Future Research

In this research a mathematical model was developed to solve the combined order picking and storage (dual command) problem of optimizing the forklift route with multiple pick and storage orders in a warehouse which has stackable pallets. We conclude that this mathematical model is a working model, as it was coded in OPL software and solved for five, seven, eight, and nine pallets size problems successfully with validated solutions. Also, two heuristics were developed which give quick solutions which are close but sub optimal to solutions out of combined model. In the future research, a Metaheuristic like genetic algorithms or simulated annealing can be used to build a model to solve bigger problems such as a hundred pallet problem.

CHAPTER FIVE

CONCLUSIONS

In the first part of the research we addressed the order picking and storage problems with stackable pallets. In this research three mathematical models were developed. One model analyzed the order picking operation in terms of travel distance. A second model analyzed the order picking operation in terms of time, adding the time elements of picking and stacking a pallet and a third model analyzed the order storage operation in terms of time. The concept of pallet stackabilty was applied in all three models. We concluded that these three mathematical models are working validated models, as they were coded in OPL software and solved for 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 pallets size problems successfully. Each was solved using five different variations of stackability options; total of 150 examples for the three models. Problems with a small number of pallets had the solution confirmed by complete enumeration. So, contributions of chapter 3 are: introduced stackability in the order picking problem, constructed a detailed time-based math programming model to find optimal pick or store sequence, and explicitly included the time associated with stacking pallets in a math programming model.

In the second part of the research we addressed the combined order picking and storage problem, also known as dual command with stackable pallets. In this research a mathematical model was developed to solve the combined order picking and storage (dual command) problem of optimizing the forklift route with multiple pick and storage orders in a warehouse which has stackable pallets. We concluded that this mathematical model is a working model, as it was coded in OPL software and solved for five, seven, eight, and nine pallets size problems successfully with validated solutions. Also, two heuristics were developed which give quick solutions which are close but sub optimal to solutions out of combined model. Comparing the results of the two heuristics versus the combined model, the combined model beats heuristics 1 and 2 in three out of four examples. So contributions of chapter 4 are: developed a detailed time-based math programming model for the dual command problem (allows picking and storing in the same route), developed two heuristics for the dual command problem, and proved that handling time is a significant time element 22 to 46 % of total time spent to pick and store pallets.

In the future research, a Metaheuristic like genetic algorithms or simulated annealing can be used to build a model to solve bigger problems such as a hundred pallet problem.

CHAPTER SIX

REFERENCES

1. BEA White Paper, “RFID for Retail: Blueprints for Bottom-Line Benefits.” 2006. Web. 10 November 2010. <http://reference.kfupm.edu.sa/content/b/e/bea_white_paper_rfid_for_retail__bluepri_77572.pdf >

2. LeBlanc, Rick. “Pallet Evolved Along with Forklift.” 2 December 2002. Web. 26 August. 2010. < http://www.palletenterprise.com/articledatabase/view.asp?articleID=821>.

3. Goetschalckx, M., and R. Wei. *Bibliography on order picking systems* 1 (2005): 1985–1992. Web. 17 August. 2010. <http://www.isye.gatech.edu/people/faculty/Marc_Goetschalckx/research.html>

4. Bartholdi, J.J. and S.T. Hackman, *Warehouse and Distribution Science.* Release 0.80. (2006), Web. 17 August. 2010. <http://www.warehouse-science.com>.

5. Coyle, J.J., E.J. Bardi and C.J. Langley, The Management of Business Logistics. (1996) St. Paul, West. Print.

6. Gu, J., M. Goetschalckx, and L. F. McGinnis, “Research on warehouse operation: A comprehensive review,” *European Journal of Operational Research* 177 (2007), 1–21. Print.

7. De Koster, R., T. Le-Duc, and K.J. Roodbergen, “Design and control of warehouse order picking: A literature review,” *European Journal of Operational Research* 182 (2007): 481–501. Print.

8. Van den Berg, J.P., and W.H.M. Zijm, “Models for warehouse management: Classification and examples,” *International Journal of Production Economics* 59 (1999): 519-528. Print.

9. Steudel, H.J. “Generating pallet loading patterns with considerations of item stacking on end and side surfaces,” *Journal of Manufacturing Systems* 3 (1984): 135-143. Print.

10. Ratliff, H.D., and A.S. Rosenthal, "Order-picking in a rectangular warehouse: A solvable case of the traveling salesman problem," *Operations Research* 31 (1983): 507–521. Print.

11. Goetschalckx, M., and H.D. Ratliff, "Order Picking In An Aisle," *IIE Transactions* 20 (1988): 53 – 62. Print.

12. De Koster, R., and E.S. Van der Poort, "Routing orderpickers in a warehouse: A comparison between optimal and heuristic solutions," *IIE Transactions* 30 (1998): 469–480. Print.

13. Roodbergen, K.J. and R. De Koster, "Routing methods for warehouses with multiple cross aisles," International Journal of Production Research 39 (2001): 1865-1883. Print.

14. Roodbergen, K.J. and R. De Koster, "Routing Order pickers in a warehouse with a middle aisle", *European Journal of Operational Research* 133 (2001): 32-43. Print.

15. Daniels, R., J. Rummel, and R. Schantz, "A model for warehouse order picking," *European Journal of Operational Research* 105 (1998): 1-17. Print.

16. Theys, C., O. Bräysy, W. Dullaert, and B. Raa, "Using a TSP heuristic for routing order pickers in warehouses," *European Journal of Operational Research* 200 (2010): 755–763. Print.

17. Hall, R.W. "Distance approximations for routing manual pickers in a warehouse," *IIE Transactions* 25 (1993): 76- 87. Print.

18. Caron, F., G. Marchet and A. Perego, "Routing policies and COI-based storage policies in picker-to-part systems," *International Journal of Production Research*. 36 (1998): 713- 732. Print.

19. Tsai, C., J. Liou, and T. Huang, "Using a multiple-GA method to solve the batch picking problem: considering travel distance and order due time," *International Journal of Production Research* 46 (2008): 6533-6555. Print.

20. Le-Duc, T., and R. De Koster, "Travel distance estimation and storage zone optimization in a 2-block class-based storage strategy warehouse," *International Journal of Production Research* 43 (2005): 3561-3581. Print.

21. Hwang, H., and G. Cho, "A performance evaluation model for order picking warehouse design," *Computers & Industrial Engineering* 51 (2006): 335–342. Print.

22. Gray, A., U. Karmarkar, and A. Seidmann, “Design and operation of an order-consolidation warehouse: Models and application,” *European Journal of Operational Research* 58 (1992): 14-36. Print.

23. Roodbergen, K. and I. Vis, “A model for warehouse layout,” *IIE Transactions*. 38 (2006): 799 – 811. Print.

24. Rana, K. “Order picking in narrow-aisle warehouses,” *International Journal of Physical Distribution and Logistics Management* 20 (1990): 9-15. Print.

25. Manzini, R., M. Gamberi, A. Persona, and A. Regattieri, “Design of a class based storage picker to product order picking system,” *International Journal of Advanced Manufacturing Technology* 32 (2007): 811–821. Print.

26. Hwang, H., and J. Song, “Sequencing picking operations and travel time models for man-on-board storage and retrieval warehousing system,” *International Journal of Production Economics* 29 (1993): 75-88. Print.

27. Chew, E.P., and L.C. Tang, “Travel time analysis for general item location assignment in a rectangular warehouse,” *European Journal of Operational Research* 112 (1999): 582-597. Print.

28. Le-Duc, T., and R. De Koster, “Travel time estimation and order batching in a 2-block warehouse,” *European Journal of Operational Research* 176 (2007): 374-388. Print.

29. Yu, M. and R. De Koster, “The impact of order batching and picking area zoning on order picking system performance,” *European Journal of Operational Research* 198 (2009): 480-490. Print.

30. Parikh, P., and R. Meller, “A Travel-Time Model for a Person-Onboard Order Picking System,” *European Journal of Operational Research* 200 (2010): 385-394. Print.

31. Queirolo, F., F. Tonelli, M. Schenone, P. Nan, and I. Zunino, “Warehouse layout design: minimizing travel time with a genetic and simulative approach methodology and case study,” *Simulation in Industry. 14th European Simulation Symposium* (2002): 271-275. Print.

32. Graves, S.C., W.H. Hausman, and L.B. Schwarz, “Storage -Retrieval Interleaving in Automatic Warehousing Systems,” *Management Science* 23(9) (1977): 935-945. Print.

33. Eynan, A. and M.J. Rosenblatt, “An interleaving policy in automated storage/retrieval systems,” *International Journal of Production Research* 31(1) (1993): 1-18. Print.

34. Lee, H.F. and S.K. Schaefer, “Sequencing methods for automated storage and retrieval systems with dedicated storage,” *Computers and Industrial Engineering* 32(2) (1997): 351-362. Print.

35. Pohl, L.M., R.D. Meller, and K. R. Gue, “An Analysis of Dual-Command Operations in Common Warehouse Designs,” *Transportation Research Part E: Logistics and Transportation Review* 45 (2009): 367-379. Print.

36. Kim, K.Y. and K.H. Kim, “A routing algorithm for a single straddle carrier to load export containers onto a containership,” *International Journal of Production Economics* 59 (1999): 425- 433. Print.

37. Brynze´r, H., and M.I. Johansson, “Design and performance of kitting and order picking systems,” *International Journal of Production Economics* 41 (1995): 115–125. Print.

38. Brynze´r, H., and M.I. Johansson, “Storage location assignment: using the product structure to reduce order picking times,” *International Journal of Production Economics* 46 (1996): 595–603. Print.

39. Caron, F., G. Marchet, and A. Perego, “Routing policies and COI-based storage policies in picker-to-part systems,” *International Journal of Production Research* 36 (3) (1998): 713–732. Print.

40. Caron, F., G. Marchet, and A. Perego, “Optimal layout in low-level picker-to-part systems,” *International Journal of Production Research* 38 (1) (2000): 101–117. Print.

41. Choe, K., and G.P. Sharp, “Small parts order picking: design and operation,” Web. 17 August 2010, <http://www.isye.gatech.edu/logisticstutorial/order/article.htm>.

42. Choe, K., G.P. Sharp, and R.S. Serfozo, “Aisle-based order pick systems with batching, zoning and sorting,” *In: Progress in Material Handling Research* (1992): 245–276. Print.

43. Clarke, G., and W. Wright, “Scheduling of vehicles from a central depot to a number of delivery points,” *Operations Research* 12 (1964): 568–581. Print.

44. Dekker, R., R. De Koster, K.J. Roodbergen, and H. Van Kalleveen, "Improving order-picking response time at Ankor's warehouse," *Interfaces* 34 (4) (2004): 303–313. Print.

45. De Koster, R., "Performance approximation of pick-to-belt orderpicking systems," *European Journal of Operational Research* 72 (1994): 558–573. Print.

46. Frazelle, E.A., and G.P. Sharp, "Correlated assignment strategy can improve order-picking operation," *Industrial Engineering* 4 (1989): 33–37. Print.

47. Goetschalckx, M., and J. Ashayeri, "Classification and design of order picking systems," *Logistics World* June, (1989): 99–106. Print.

48. Goetschalckx, M., and D. H. Ratliff, "An efficient algorithm to cluster order picking items in a wide aisle," *Engineering Costs and Production Economy* 13 (1988a): 263–271. Print.

49. Hwang, H., Y.H. Oh, and Y.K. Lee, "An evaluation of routing policies for order-picking operations in low-level picker-to part system," *International Journal of Production Research* 42 (18) (2004): 3873–3889. Print.

50. Jane, C.C., and Y.W. Laih, "A clustering algorithm for item assignment in a synchronized zone order picking system," *European Journal of Operational Research* 166 (2) (2005): 489–496. Print.

51. Le-Duc, T., "Design and control of efficient order picking processes," *Ph.D. thesis, RSM Erasmus University* 2005. Print.

52. Petersen, C.G., "An evaluation of order picking routing policies," *International Journal of Operations & Production Management* 17 (11) (1997): 1098–1111. Print.

53. Rouwenhorst, B., B. Reuter, V. Stockrahm, G.J. Van Houtum, R.J. Mantel, and W.H.M. Zijm, "Warehouse design and control: framework and literature review," *European Journal of Operational Research* 122 (2000): 515–533. Print.

54. Sarker, B.R., and P.S. Babu, "Travel time models in automated storage/retrieval systems: A critical review," *International Journal of Production Economics* 40 (1995): 173–184. Print.

55. Tang, L.C., and E.P. Chew, "Order picking systems: batching and storage assignment strategies," *Computer & Industrial Engineering* 33 (3) (1997): 817–820. Print.

56. Wa¨scher, G., "Order picking: A survey of planning problems and methods," *In: Supply Chain Management and Reverse Logistics* (2004): 323–347. Print.

57. Hassan Aly, A. and W. Ferrell, "Order-picking path optimization with multiple picks per route using pallet stackability concept in a warehouse," *Proceedings of the Industrial Engineering Research Conference* (2009). Print.

58. Petersen, C.G. and G. Aase, "A comparison of picking, storage, and routing policies in manual order picking," *International Journal of Production Economics* 92 (2004): 11-19. Print.

59. Hassan Aly, A., "Modeling Order Loading Using Pallet Stackability Concept in a Production Warehouse to Minimize Total Travel Time," *Proceedings of the 7th International Scientific Conference on Production Engineering, RIM* September (2009). Print.

60. Vollmann, T. and E. Buffa, "The Facilities Layout Problem in Perspective," *Management Science* 12 (1966): 450-468. Print.

61. Malmborg, C.J., and K. Al-Tassan, "An integrated performance model for orderpicking systems with randomized storage," *Applied Mathematical Modelling*, 24 (2) (2000): 95-111. Print.

62. Bozer, Y.A., and J.A. White, "Travel-time Models for Automated Storage/Retrieval Systems," *IIE Transactions (Institute of Industrial Engineers)* 16 (4) (1984): 329-338. Print.

APPENDICES

Appendix A: 50 pallets distance matrix

D=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26
D	0	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	50
P1	290	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290
P2	280	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280
P3	270	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270
P4	260	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260
P5	250	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250
P6	240	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240
P7	230	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230
P8	220	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220
P9	210	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210
P10	200	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200
P11	190	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190
P12	180	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180
P13	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170
P14	160	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160
P15	150	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150
P16	140	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140
P17	130	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130
P18	120	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120
P19	110	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110
P20	100	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100
P21	90	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90

D=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26
P22	80	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80
P23	70	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70
P24	60	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60
P25	50	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50
P26	50	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0
P27	60	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50
P28	70	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60
P29	80	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70
P30	90	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80
P31	100	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90
P32	110	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100
P33	120	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110
P34	130	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120
P35	140	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130
P36	150	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150	140
P37	160	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160	150
P38	170	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170	160
P39	180	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180	170
P40	190	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190	180
P41	200	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200	190
P42	210	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210	200

D=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26
P43	220	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220	210
P44	230	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230	220
P45	240	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240	230
P46	250	490	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250	240
P47	260	500	490	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260	250
P48	270	510	500	490	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270	260
P49	280	520	510	500	490	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280	270
P50	290	530	520	510	500	490	480	470	460	450	440	430	420	410	400	390	380	370	360	350	340	330	320	310	300	290	280
Dummy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

D=	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
D	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	0
P1	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	490	500	510	520	530	0
P2	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	490	500	510	520	0
P3	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	490	500	510	0
P4	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	490	500	0
P5	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	490	0
P6	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	480	0
P7	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	470	0
P8	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	460	0
P9	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	450	0
P10	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	440	0
P11	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	430	0
P12	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	420	0
P13	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	410	0
P14	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	400	0
P15	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	390	0
P16	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	380	0
P17	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	370	0
P18	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	360	0
P19	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	350	0
P20	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	340	0
P21	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	330	0

D=	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
P22	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	320	0
P23	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	310	0
P24	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	300	0
P25	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	290	0
P26	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	280	0
P27	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	270	0
P28	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	260	0
P29	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	250	0
P30	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	240	0
P31	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	230	0
P32	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	220	0
P33	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210	0
P34	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	0
P35	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	0
P36	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	180	0
P37	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	170	0
P38	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	160	0
P39	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	150	0
P40	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	140	0
P41	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	130	0
P42	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	110	120	0

D=	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
P43	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	5	0	50	60	70	80	90	100	110	0
P44	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	100	0
P45	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	90	0
P46	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	80	0
P47	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	70	0
P48	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	60	0
P49	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	50	0
P50	270	260	250	240	230	220	210	200	190	180	170	160	150	140	130	120	110	100	90	80	70	60	50	0	0
Dummy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

50 pallets stackabilty matrix

S=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27
D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P6	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P7	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P8	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P9	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P10	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P11	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P12	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
P14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
P16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
P17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
P18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
P19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
P20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
P21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
P22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

S=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27
P23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
P24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
P25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
P26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
P27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

S=	D	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27
P46	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dummy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

S=	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

S=	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
P23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P27	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P28	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P29	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P30	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P31	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P32	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P33	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P34	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P35	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P36	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P37	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
P38	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
P39	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
P40	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
P41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
P42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

S=	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	Dummy
P43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
P44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
P45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
P46	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
P47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
P48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
P49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
P50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Dummy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Appendix B: 5 pallets stackability and distance matrices

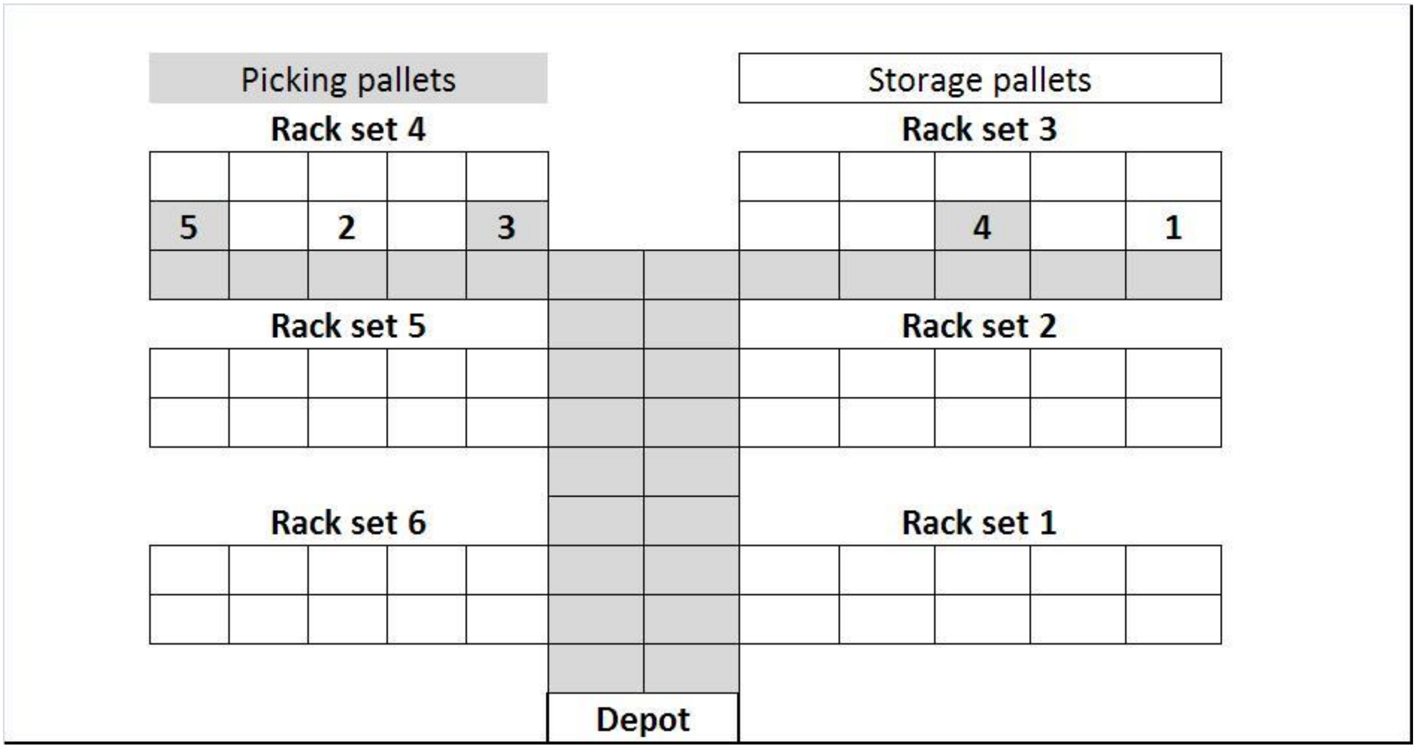
Distance Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	Dummy
Depot	0	70	60	50	60	70	0
pallet 1	70	0	45	35	10	55	0
pallet 2	60	45	0	10	35	10	0
pallet 3	50	35	10	0	25	20	0
pallet 4	60	10	35	25	0	45	0
pallet 5	70	55	10	20	45	0	0
Dummy	0	0	0	0	0	0	0

Stackability Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	Dummy
Depot	0	0	0	0	0	0	0
pallet 1	0	0	1	0	0	0	0
pallet 2	0	0	0	1	0	1	0
pallet 3	0	0	0	0	1	0	0
pallet 4	0	0	0	0	0	1	0
pallet 5	0	0	0	0	0	0	0
Dummy	0	0	0	0	0	0	0

Layout

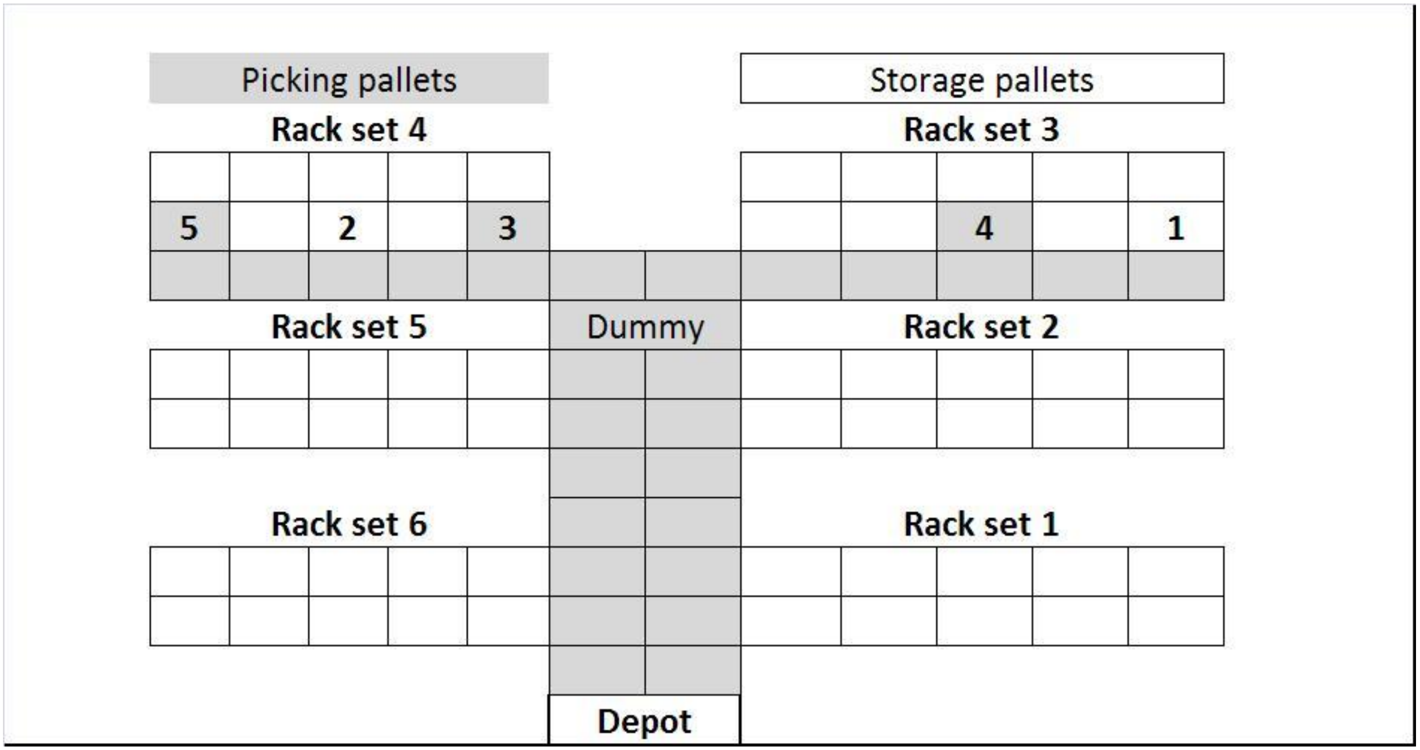


Picking pallets
Storage pallets
Rack set 4
Rack set 3
5
2
3
4
1
Rack set 5
Rack set 2
Rack set 6
Rack set 1
Depot

Distance Matrix with dummy for Heuristic 2

	Depot	pallet 1	pallet 2	Dummy	pallet 3	pallet 4	pallet 5
Depot	0	70	60	35	50	60	70
pallet 1	70	0	45	30	35	10	55
pallet 2	60	45	0	20	10	35	10
Dummy	35	30	20	0	10	20	30
pallet 3	50	35	10	10	0	25	20
pallet 4	60	10	35	20	25	0	45
pallet 5	70	55	10	30	20	45	0

Layout with dummy for Heuristic 2



Appendix C: 7 pallets stackability and distance matrices

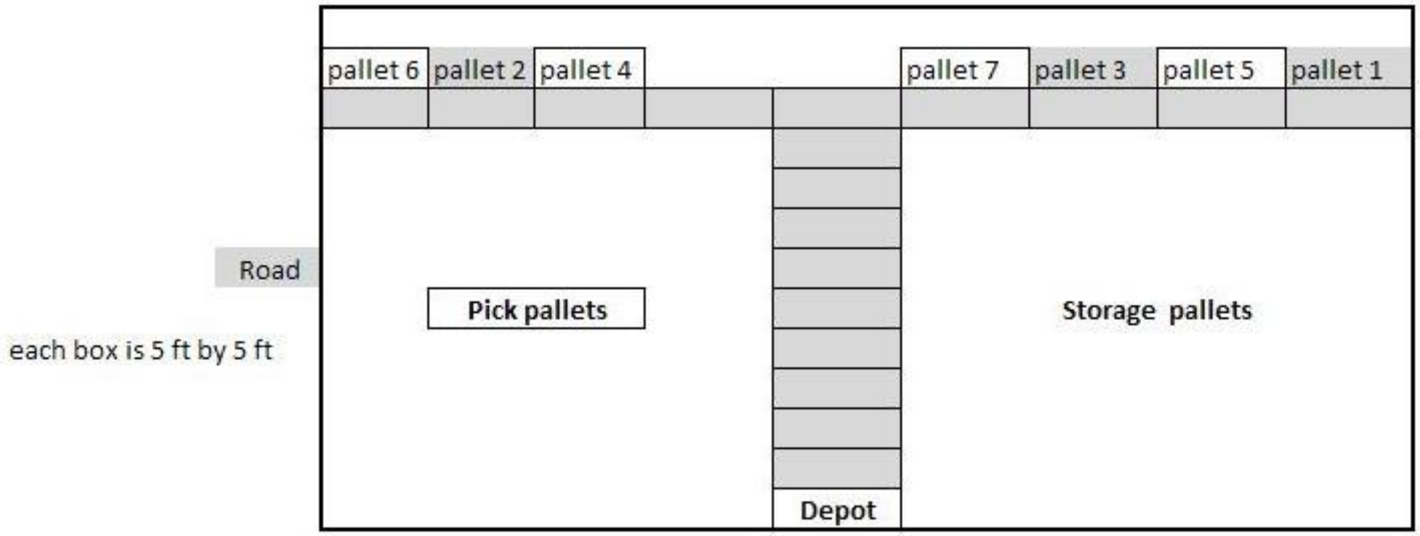
Distance Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	Dummy
Depot	0	70	65	60	60	65	70	55	0
pallet 1	70	0	35	10	30	5	40	15	0
pallet 2	65	35	0	25	5	30	5	20	0
pallet 3	60	10	25	0	20	5	30	5	0
pallet 4	60	30	5	20	0	25	10	15	0
pallet 5	65	5	30	5	25	0	35	10	0
pallet 6	70	40	5	30	10	35	0	25	0
pallet 7	55	15	20	5	15	10	25	0	0
Dummy	0	0	0	0	0	0	0	0	0

Stackability Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	Dummy
Depot	0	0	0	0	0	0	0	0	0
pallet 1	0	0	0	1	0	0	0	1	0
pallet 2	0	0	0	0	0	1	0	0	0
pallet 3	0	1	0	0	0	0	0	0	0
pallet 4	0	0	0	0	0	0	1	0	0
pallet 5	0	0	1	0	0	0	0	0	0
pallet 6	0	0	0	0	1	0	0	0	0
pallet 7	0	0	0	0	0	0	0	0	0
Dummy	0	0	0	0	0	0	0	0	0

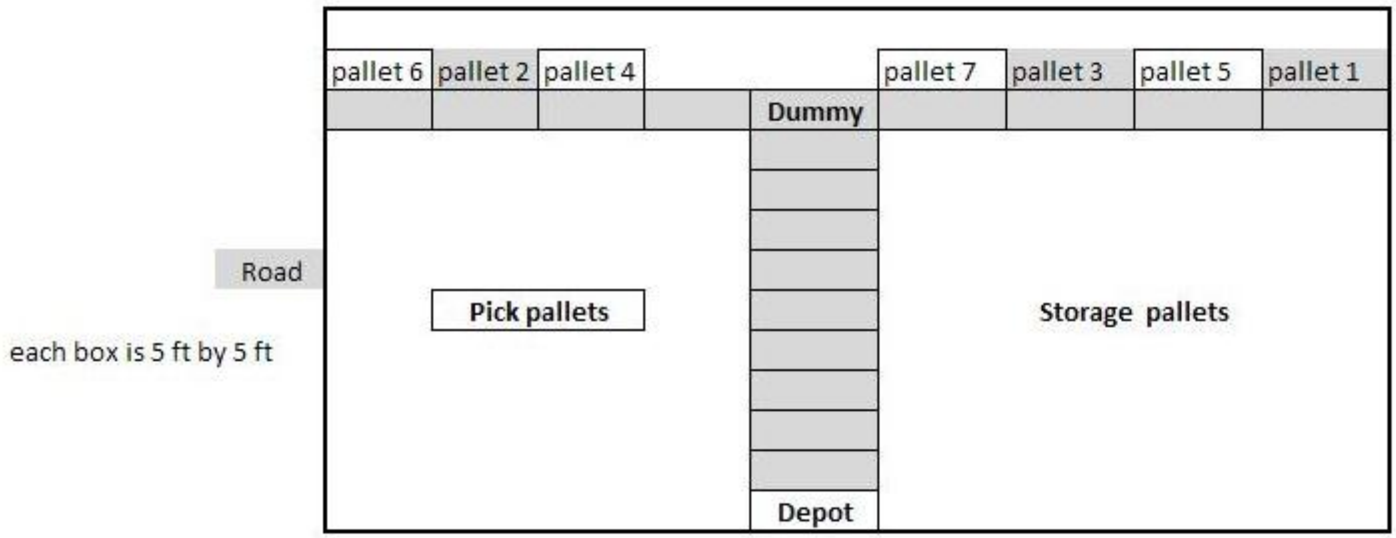
Layout


pallet 6
pallet 2
pallet 4
pallet 7
pallet 3
pallet 5
pallet 1
Road
Pick pallets
Storage pallets
each box is 5 ft by 5 ft
Depot

Distance Matrix with dummy for Heuristic 2

	Depot	pallet 1	pallet 2	pallet 3	Dummy	pallet 4	pallet 5	pallet 6	pallet 7
Depot	0	70	65	60	45	60	65	70	55
pallet 1	70	0	35	10	20	30	5	40	15
pallet 2	65	35	0	25	15	5	30	5	20
pallet 3	60	10	25	0	10	20	5	30	5
Dummy	45	20	15	10	0	10	15	20	5
pallet 4	60	30	5	20	10	0	25	10	15
pallet 5	65	5	30	5	15	25	0	35	10
pallet 6	70	40	5	30	20	10	35	0	25
pallet 7	55	15	20	5	5	15	10	25	0

Layout with Dummy



Appendix D: 8 scattered pallets stackability and distance matrices

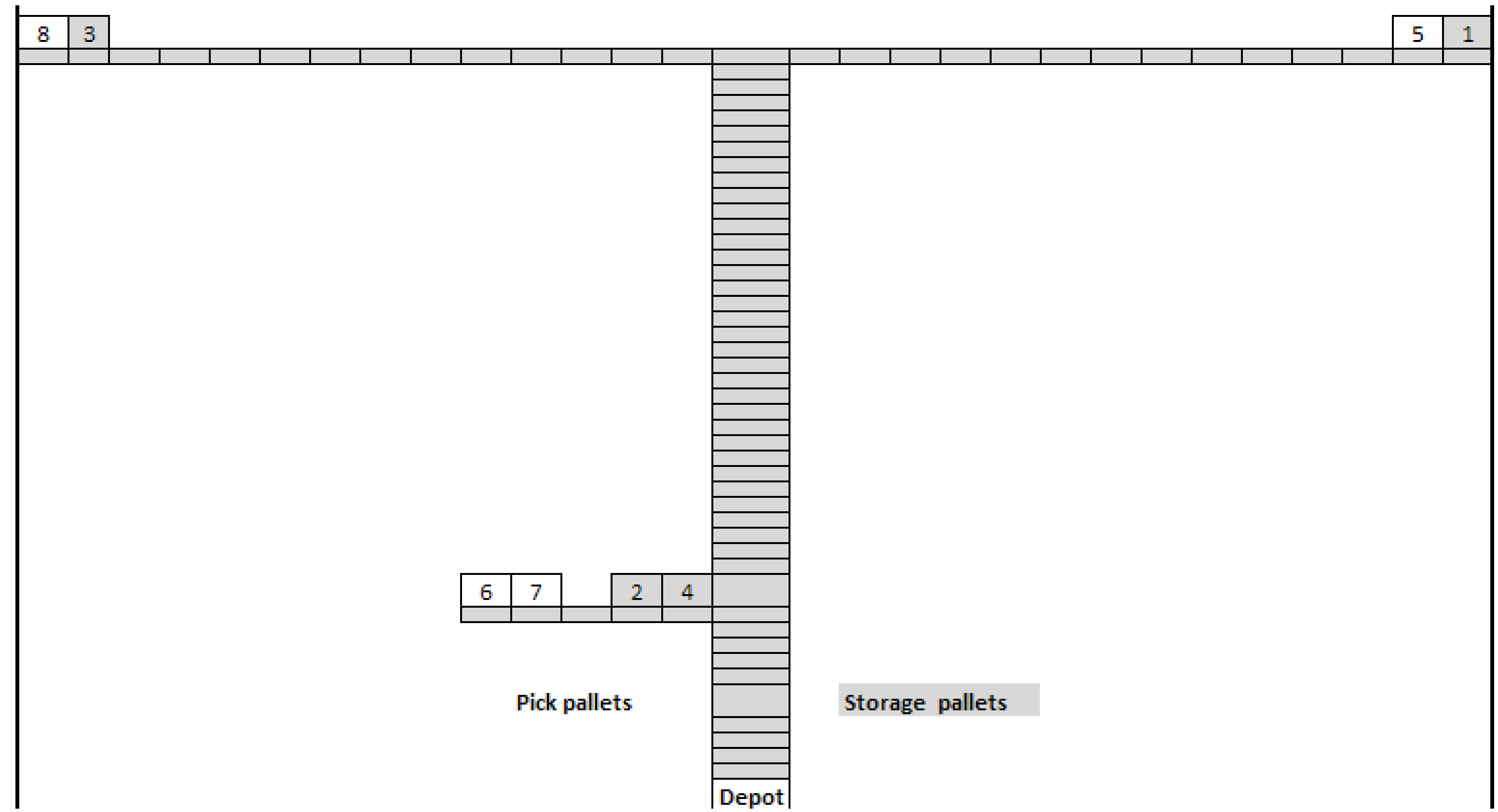
Distance Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	Dummy
Depot	0	295	60	290	55	290	75	70	295	0
pallet 1	295	0	255	135	250	5	270	265	140	0
pallet 2	60	255	0	250	5	250	15	10	255	0
pallet 3	290	135	250	0	245	130	265	260	5	0
pallet 4	55	250	5	245	0	245	20	15	250	0
pallet 5	290	5	250	130	245	0	265	260	135	0
pallet 6	75	270	15	265	20	265	0	5	270	0
pallet 7	70	265	10	260	15	260	5	0	265	0
pallet 8	295	140	255	5	250	135	270	265	0	0
Dummy	0	0	0	0	0	0	0	0	0	0

Stackability Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	Dummy
Depot	0	0	0	0	0	0	0	0	0	0
pallet 1	0	0	1	1	1	1	1	1	1	0
pallet 2	0	1	0	1	1	1	1	1	1	0
pallet 3	0	1	1	0	1	1	1	1	1	0
pallet 4	0	1	1	1	0	1	1	1	1	0
pallet 5	0	1	1	1	1	0	1	1	1	0
pallet 6	0	1	1	1	1	1	0	1	1	0
pallet 7	0	1	1	1	1	1	1	0	1	0
pallet 8	0	1	1	1	1	1	1	1	0	0
Dummy	0	0	0	0	0	0	0	0	0	0

Layout



Distance Matrix with Dummy for Heuristic 2

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	dummy	pallet 5	pallet 6	pallet 7	pallet 8
Depot	0	295	60	290	55	145	290	75	70	295
pallet 1	295	0	255	135	250	155	5	270	265	140
pallet 2	60	255	0	250	5	110	250	15	10	255
pallet 3	290	135	250	0	245	150	130	265	260	5
pallet 4	55	250	5	245	0	105	245	20	15	250
dummy	145	155	110	150	105	0	150	125	120	155
pallet 5	290	5	250	130	245	150	0	265	260	135
pallet 6	75	270	15	265	20	125	265	0	5	270
pallet 7	70	265	10	260	15	120	260	5	0	265
pallet 8	295	140	255	5	250	155	135	270	265	0

Appendix E: 9 pallets stackability and distance matrices

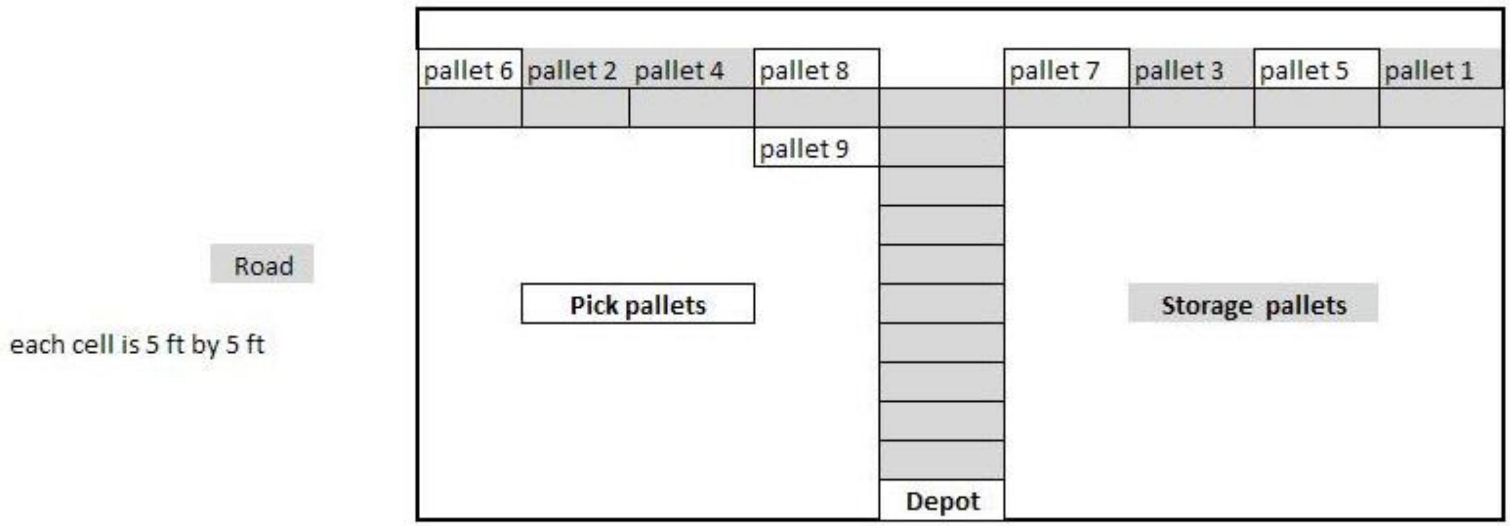
Distance Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	pallet 9	Dummy
Depot	0	70	65	60	60	65	70	55	55	45	0
pallet 1	70	0	35	10	30	5	40	15	25	25	0
pallet 2	65	35	0	25	5	30	5	20	10	10	0
pallet 3	60	10	25	0	20	5	30	5	15	15	0
pallet 4	60	30	5	20	0	25	10	15	5	5	0
pallet 5	65	5	30	5	25	0	35	10	20	20	0
pallet 6	70	40	5	30	10	35	0	25	15	15	0
pallet 7	55	15	20	5	15	10	25	0	10	10	0
pallet 8	55	25	10	15	5	20	15	10	0	5	0
pallet 9	45	25	10	15	5	20	15	10	5	0	0
Dummy	0	0	0	0	0	0	0	0	0	0	0

Stackability Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	pallet 9	Dummy
Depot	0	0	0	0	0	0	0	0	0	0	0
pallet 1	0	0	0	1	0	0	0	1	0	0	0
pallet 2	0	0	0	0	0	1	0	0	0	0	0
pallet 3	0	1	0	0	0	0	0	0	1	0	0
pallet 4	0	0	0	0	0	0	1	0	0	0	0
pallet 5	0	0	1	0	0	0	0	0	0	0	0
pallet 6	0	0	0	0	1	0	0	0	0	0	0
pallet 7	0	0	0	0	0	0	0	0	0	1	0
pallet 8	0	0	0	0	0	0	0	0	0	0	0
pallet 9	0	0	0	0	0	0	0	0	0	0	0
Dummy	0	0	0	0	0	0	0	0	0	0	0

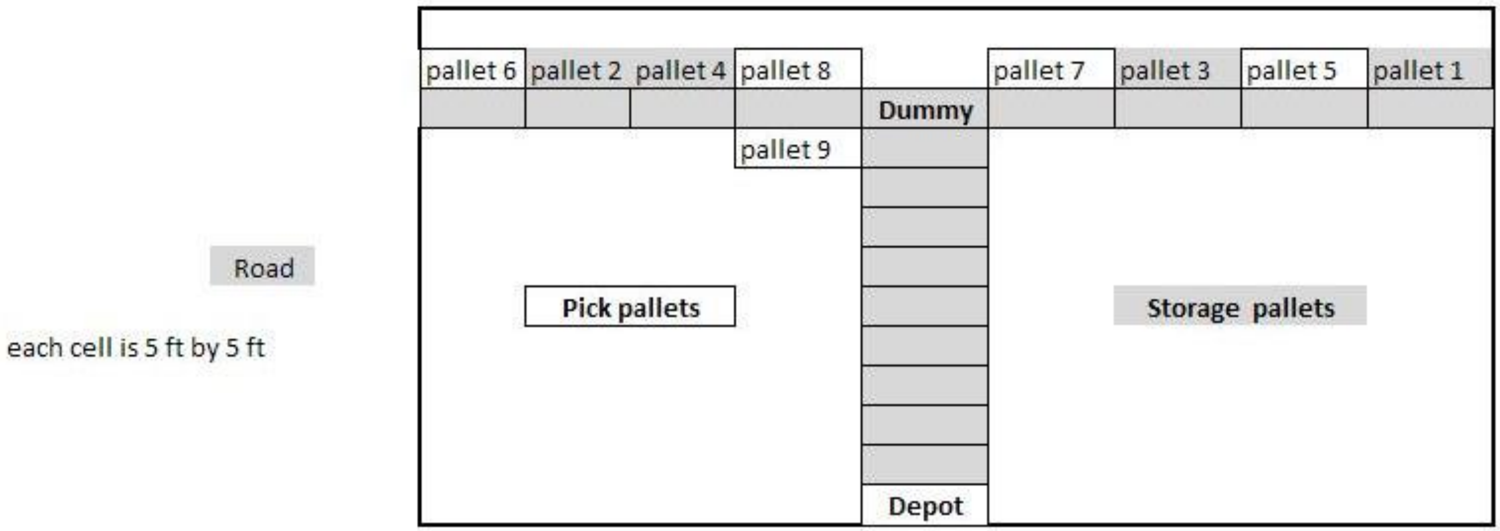
Layout


Road
each cell is 5 ft by 5 ft
pallet 6
pallet 2
pallet 4
pallet 8
pallet 7
pallet 3
pallet 5
pallet 1
pallet 9
Pick pallets
Storage pallets
Depot

Distance Matrix with dummy for Heuristic 2

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	Dummy	pallet 5	pallet 6	pallet 7	pallet 8	pallet 9
Depot	0	70	65	60	60	45	65	70	55	55	45
pallet 1	70	0	35	10	30	20	5	40	15	25	25
pallet 2	65	35	0	25	5	15	30	5	20	10	10
pallet 3	60	10	25	0	20	10	5	30	5	15	15
pallet 4	60	30	5	20	0	10	25	10	15	5	5
Dummy	45	20	15	10	10	0	15	20	5	5	5
pallet 5	65	5	30	5	25	15	0	35	10	20	20
pallet 6	70	40	5	30	10	20	35	0	25	15	15
pallet 7	55	15	20	5	15	5	10	25	0	10	10
pallet 8	55	25	10	15	5	5	20	15	10	0	5
pallet 9	45	25	10	15	5	5	20	15	10	5	0

Layout with Dummy for Heuristic 2



Appendix F: 8 close by pallets stackability and distance matrices (Used in Figure 4.3)

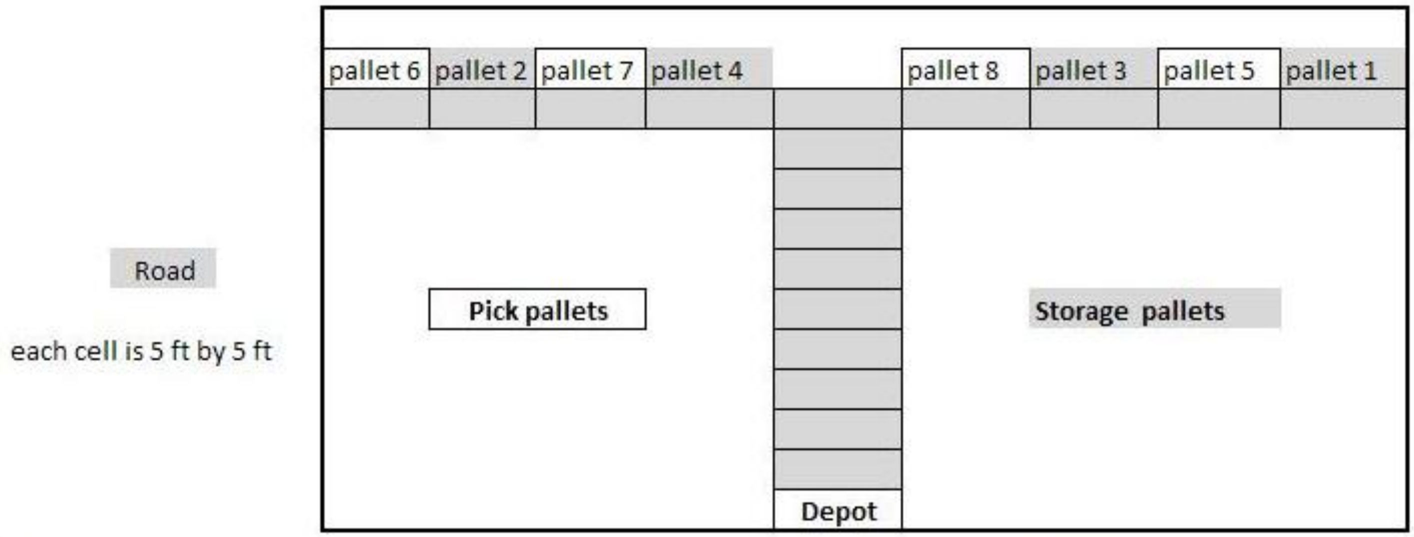
Distance Matrix

	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	Dummy
Depot	0	70	65	60	55	65	70	60	55	0
pallet 1	70	0	35	10	25	5	40	30	15	0
pallet 2	65	35	0	25	10	30	5	5	20	0
pallet 3	60	10	25	0	15	5	30	20	5	0
pallet 4	55	25	10	15	0	20	15	5	10	0
pallet 5	65	5	30	5	20	0	35	25	10	0
pallet 6	70	40	5	30	15	35	0	10	25	0
pallet 7	60	30	5	20	5	25	10	0	15	0
pallet 8	55	15	20	5	10	10	25	15	0	0
Dummy	0	0	0	0	0	0	0	0	0	0

Stackability Matrix

S=	Depot	pallet 1	pallet 2	pallet 3	pallet 4	pallet 5	pallet 6	pallet 7	pallet 8	Dummy
Depot	0	0	0	0	0	0	0	0	0	0
pallet 1	0	0	1	0	0	0	0	0	0	0
pallet 2	0	1	0	1	0	0	0	0	0	0
pallet 3	0	0	1	0	1	0	1	0	0	0
pallet 4	0	0	0	1	0	1	0	1	0	0
pallet 5	0	0	0	0	1	0	1	0	1	0
pallet 6	0	0	0	1	0	1	0	1	0	0
pallet 7	0	0	0	0	1	0	1	0	1	0
pallet 8	0	0	0	0	0	1	0	1	0	0
Dummy	0	0	0	0	0	0	0	0	0	0

Layout



pallet 6
pallet 2
pallet 7
pallet 4
pallet 8
pallet 3
pallet 5
pallet 1
Road
each cell is 5 ft by 5 ft
Pick pallets
Storage pallets
Depot