Obtaining optimal and approximate solutions to the problem of scheduling inbound and outbound trucks in cross docking operations

SHAHIN NOURMOHAMMADI SHARABIANI

The Thesis comprises 30 ECTS credits and is a compulsory part in the Master of Science in Industrial Management with specialization in Logistics, 120 ECTS Nr 12/2009.
Obtaining optimal and approximate solutions to the problem of scheduling inbound and outbound trucks in cross docking operations

SHAHIN NOURMOHAMMADI SHARABIANI

Master thesis

Subject Category: Technology

University of Borås
School of Engineering
SE-501 90 BORÅS
Telephone +46 033 435 4640

Examiner: Göran Stjernman
Supervisor name: Associate Professor. Dr. Peruvemba Sundaram Ravi
Supervisor address: School Of Business & Economics, Wilfrid Laurier University
Waterloo, Ontario N2L3C5, Canada

Client:

Keywords: Cross docking; Heuristics; Logistics; Mathematical modeling; Optimization; Truck scheduling
Acknowledgment

I would like to thank my supervisor, Associate Professor Dr. Peruvemba Sundaram Ravi from School of Business & Economics, Wilfrid Laurier University (Waterloo, Ontario, Canada). Firstly for his kindness in accepting to be my supervisor and secondly, for his invaluable assistance, support and guidance.
I would also like to thank teachers and staff at University of Boras for their help and support during my studies.
Dedication

To my family, especially to my mother, for their love, endless support and encouragement.

To Ivanka, my best friend.
Abstract

The thesis focuses on optimization of inbound and outbound truck scheduling with the goal of minimizing total operation time of cross docking. A model of cross docking is developed; two different methods are applied on the model in order to find an optimal docking sequence for receiving and shipping trucks and their assignment to receiving and shipping docks, and product routing from receiving to shipping trucks. The two methods used were mathematical modeling and heuristic algorithm. For the first method, a mixed integer programming model was developed to minimize total operation time; AMPL modeling language is used for the mathematical modeling for small sized problems. For the second method, a heuristic algorithm was developed to find near optimal solutions fast and was used for problems of larger size. In order to examine the performance of heuristic algorithm, small problems were solved by both mathematical model and the heuristic algorithm. The results from the mathematical model and the heuristic algorithm are very close with slight differences in receiving and shipping truck docking sequence, and in product routing between these two methods. In addition, the heuristic algorithm also calculates number of products transferring from receiving trucks to the temporary storage as well as the number of products transferring from the temporary storage to shipping truck in contrary to the mathematical model. Total number of units of products passing through the temporary storage calculated by heuristic algorithm is presented and it can be seen that the heuristic algorithm transfers to the temporary storage as few products as possible. Furthermore, in cases that receiving and shipping trucks are divided into groups or clusters in the cross docking operation, heuristic algorithm can be used to calculate optimal number of receiving and shipping docks based on preferences of total operation time or total number of products passing through the temporary storage. Another issue which is focused on is the problem of dock door assignment. Close shipping docks to each receiving dock are determined and the percentage of products transferred from a receiving dock to its close shipping docks is calculated as a method to measure the performance of the dock assignment solution.

Keywords: Cross docking; Heuristics; Logistics; Mathematical modeling; Optimization; Truck scheduling
Table of contents

List of figures ........................................................................................................................................ viii
List of tables ........................................................................................................................................... ix
1 Introduction ......................................................................................................................................... 1
  1.1 Introduction to cross docking .................................................................................................. 1
  1.2 Cross docking implementation ............................................................................................ 2
  1.3 Research objectives ........................................................................................................... 3
  1.4 Organization of the thesis ................................................................................................. 4
2 Literature review .............................................................................................................................. 5
  2.1 Shape of cross dock facility ................................................................................................ 5
  2.2 Dock door assignment problem ........................................................................................ 5
  2.3 Positioning of goods in temporary storage ...................................................................... 6
  2.4 Truck scheduling ............................................................................................................... 6
  2.5 Miscellaneous .................................................................................................................... 8
3 Model description ............................................................................................................................. 10
  3.1 Assumptions ....................................................................................................................... 11
  3.2 Expected results .............................................................................................................. 12
  3.3 Model development ......................................................................................................... 12
4 Mathematical model ......................................................................................................................... 14
  4.1 Notations ............................................................................................................................ 14
  4.2 Mixed integer programming model ............................................................................... 16
  4.3 Implementation ................................................................................................................ 19
5 Heuristic method .............................................................................................................................. 20
  5.1 Notations ............................................................................................................................ 20
  5.2 Heuristic algorithm of the scheduling problem .............................................................. 22
    5.2.1 Receiving and shipping dock assignment ................................................................ 25
    5.2.2 Truck timing ............................................................................................................. 25
    5.2.3 Scheduling procedure .............................................................................................. 26
    5.2.4 Procedure 1 – Update product transfer .................................................................. 33
    5.2.5 Procedure 2 – Finding a pair of unscheduled trucks .............................................. 35
    5.2.6 Procedure 3 - Finding a receiving truck to schedule .............................................. 38
    5.2.7 Procedure 4 - Finding a shipping truck to schedule .............................................. 41
    5.2.8 Procedure 5 – Finding product transfer between docking trucks ..................... 43
    5.2.9 Procedure 6 – Receiving dock assignment ............................................................. 46
    5.2.10 Procedure 7 – Shipping dock assignment ............................................................ 48
    5.2.11 Procedure 8 – Truck timing .................................................................................. 50
    5.2.12 Procedure 9 – Product transfer from temporary storage .................................. 55
6 Results and conclusions ..................................................................................................................... 57
  6.1 Results of mathematical model ........................................................................................... 57
  6.2 Results of heuristic algorithm ........................................................................................... 63
  6.3 Dock assignment results ................................................................................................... 65
  6.4 Conclusions ....................................................................................................................... 69
List of references .................................................................................................................................. 71
Appendix A Mathematical model formulation in AMPL ................................................................ 73
Appendix B Data sets used in the research ................................................................................... 76
Appendix C  Scheduling algorithm coded in visual basic programming language 87
List of figures

Figure 1-1. Basic cross docking system .................................................................................. 2
Figure 2-1. Possible combinations for model generation (Reproduced from Yu, 2002, p.12) .................................................................................................................. 7
Figure 3-1. Product routing in the cross docking model ......................................................... 10
Figure 5-1. Flowchart - Scheduling algorithm ........................................................................ 30
Figure 5-2. Flowchart - Scheduling algorithm (continue) ...................................................... 31
Figure 5-3. Flowchart - Scheduling algorithm (continue) ...................................................... 32
Figure 5-4. Flowchart - Update product transfer procedure .................................................. 34
Figure 5-5. Flowchart - Finding a pair of unscheduled trucks procedure .............................. 37
Figure 5-6. Flowchart - Finding a receiving truck to schedule procedure ............................. 40
Figure 5-7. Flowchart - Finding a shipping truck to schedule procedure .............................. 42
Figure 5-8. Flowchart - Finding product transfer between docking trucks procedure .......... 45
Figure 5-9. Flowchart - Receiving dock assignment procedure .......................................... 47
Figure 5-10. Flowchart - Shipping dock assignment procedure ........................................... 49
Figure 5-11. Flowchart – Timing procedure ........................................................................ 53
Figure 5-12. Flowchart – Timing procedure (continue) ........................................................ 54
Figure 5-13. Flowchart - Product transfer from temporary storage procedure ....................... 56
Figure 6-1. Sample layout of receiving and shipping docks .................................................. 65
Figure C-1. Scheduling program - snapshot (Main page) ...................................................... 99
Figure C-2. Scheduling program - snapshot (Data input page 1) ........................................... 99
Figure C-3. Scheduling program - snapshot (Data input page 2) .......................................... 100
Figure C-4. Scheduling program - snapshot (Data input page 3) .......................................... 101
Figure C-5. Scheduling program - snapshot (Output page 1) ................................................. 101
Figure C-6. Scheduling program - snapshot (Output page 2) ................................................ 102
Figure C-7. Scheduling program - snapshot (Output page 3) ................................................ 103
List of tables

Table 2-1. Three models studied by Yu (2002) ................................................................. 7
Table 2-2. Model studied in this thesis ............................................................................. 8
Table 4-1. Sample values for $p_j$ and $q_j$ in a mathematical model ............................. 19
Table 6-1. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 50. ................................................................................... 58
Table 6-2. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 100. ...................................................................................... 59
Table 6-3. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 150. ................................................................................. 60
Table 6-4. Product routing results for data sets 1 and 3 .................................................. 61
Table 6-5. Results for data set 4 with different number of docks .................................... 62
Table 6-6. Heuristic algorithm’s results for data sets 6 to 10 ........................................ 64
Table 6-7. Receiving and shipping dock door assignment results for all data sets. ...... 67
Table 6-8. Percentage of products transferred to close shipping docks from receiving docks ......................................................................................................................... 68
Table 6-9. Percentage of products transferred to close shipping docks from receiving docks (continue) .......................................................................................................... 69
Table B-1. Data set 1 ...................................................................................................... 76
Table B-2. Data set 2 ...................................................................................................... 77
Table B-3. Data set 3 ...................................................................................................... 78
Table B-4. Data set 4 ...................................................................................................... 79
Table B-5. Data set 5 ...................................................................................................... 80
Table B-6. Distances between receiving and shipping docks used for data sets 6 to 10.  81
Table B-7. Loading and unloading time for products used for data sets 6 to 10 ............. 81
Table B-8. Data set 6 .................................................................................................... 82
Table B-9. Data set 7 .................................................................................................... 83
Table B-10. Data set 8 .................................................................................................. 84
Table B-11. Data set 9 .................................................................................................. 85
Table B-12. Data set 10 ................................................................................................. 86
1 Introduction

1.1 Introduction to cross docking

Recently, many companies after achieving significant improvements in their manufacturing operations are focusing on improving their logistics and distribution operations. Therefore, managing the supply and distribution network has become very challenging. Cross docking is an innovative warehousing strategy that has a great potential for controlling the logistics and distribution costs while simultaneously maintaining the level of customer satisfaction. Cross docking is a process of moving products through distribution centers without storing them (Apte 2000). In other words, cross docking is a direct flow of products from the receiving process to the shipping process with least additional handling and storage, and it is as much information handling as it is material handling (Schwind 1995). Compared to traditional warehousing with major functions of receiving, storage, order picking, and shipping, cross docking technique eliminates the storage and order picking functions of a warehouse while still allowing it to serve its receiving and shipping functions (Bartholdi & Gue 2004s). For cross docking to work properly, there should be a balance between the incoming load and the outgoing load as products never stay in a cross docking warehouse for more than 18-24 hours. Therefore, items that come into the warehouse should be demanded or pulled by retailers or destination points quickly (Apte 2000).

Cross docking is commonly used in cases when products cannot be shipped directly. The operations which take place in the cross docking are the following:

1. loads are scheduled for delivery into the cross dock facility from vendors – the goods need to arrive according to a strict time schedule which is linked to the vehicle departure times;
2. inbound products are sorted immediately into their outbound orders – as the outgoing vehicles can be taking a mix of cross-docked goods and stocked goods as well, a great degree of coordination is required in order not to cause any unnecessary delays in uploading products;
3. these orders are moved immediately into the shipping docks.

Compared to the traditional warehousing, activities such as receiving inspection, receiving staging, storage, order picking or order assembly are omitted. (Frazelle 2001, p. 246; Rushton et al 2006, p. 321)

Figure 1-1 shows a very basic cross docking system where products from different suppliers arrive at the cross dock and after consolidation, they are sent to shipping trucks leaving to different destinations.
As there exist a number of variations of cross docking, details can vary among different companies according to their particular needs.

According to Cross-docking trends report 2008, some of the benefits of cross docking reported by companies participating in the research are:

- Improved service level;
- Reduced transportation costs;
- Reduced need for warehouse space;
- Consolidated shipments to destination;
- Reduced inventory keeping costs;
- Reduced labor costs.

Despite all the advantages of cross docking there are many issues which have to be taken into account before implementation of cross docking. Some of these issues are discussed in the next section.

### 1.2 Cross docking implementation

While implementing and operating cross dock facility, certain criteria have to be met in order to make it work; according to Schaffer (2000), this criteria can be divided into six categories as follows:

1. Partnering with other members of the distribution chain (for example in case that not all the members of the distribution chain implemented cross docking, more planning and cooperation is needed in order not to increase costs on the side of these companies);

2. Absolute confidence in the quality and availability of product (as cross docking is a real-time process, it needs products flow without interruption, which means that the cross docking operations have to be sure that the right products in required quality are available according to demand);

3. Communications between supply chain members (all information concerning cross docking operations has to be distributed to the relevant places in the cooperating companies as soon as they become available);
4. Communication and control within the cross docking operations (once products arrive at the cross dock facility, they have to move quickly and without interruption through the facility with the help of warehouse management system);

5. Personnel, equipment and facilities (before implementing cross docking, sufficient sources should be allocated to ensure that all areas are covered such as space for waiting trucks, equipment and personnel for product transfer within the facility etc.),

6. Tactical management (it is necessary that additional human resources are available when some problem arises, often in means of additional supervisory workload to the usual tasks).

Another issue in implementing cross docking systems is that even though cross docking has a number of advantages mentioned earlier, it is not suitable to be implemented in every situation. To identify when cross docking could be of no or small advantage, it is necessary to focus on the overall supply chain in the particular situation.

Among the situations when it is not always suitable to implement cross docking are according to Rushton et al (2006) for example the following:

- Considerably more handling space would be required for sorting activities within the cross dock facility;
- Product are transported in less than pallet load quantities or less than truck load quantities – implementing cross dock can lead to the increase of transport costs;
- Inventory at the warehouse would be just replaced by inventory held in the supply chain to support the just-in-time basis of cross docking – an overall view on all parts of the supply chain has to be taken so that the total inventory will be reduced after cross docking is implemented;
- Thousands of different product from large number of suppliers are circulating, and as a result of the cross docking the coordination among all members would become too complex and difficult to maintain.

As discussed in the previous and this section, cross docking is an important strategy to manage supply chains and distribution networks efficiently and cost effectively. On the other hand, it is also important to manage cross docking operation efficiently by improving cross docking processes. There exist studies concerned with different processes of cross docking aiming to improve cross docking operation. Some of these studies will be discussed in next chapter.

### 1.3 Research objectives

As cross docking is getting more popular and widely used by many companies, it is necessary to have more research conducted to improve cross docking operation. One of important operations of cross docking is to schedule receiving and shipping trucks and assign them to proper docks. Optimal schedule for incoming and outgoing trucks can decrease operation time of cross docking as well as decrease material handling inside the cross dock terminal.
In this research, the focus is on optimization of inbound and outbound truck scheduling with the goal of minimizing total operation time of cross docking. The methods used in this research to optimize truck scheduling are mathematical modeling and heuristic method. An optimal docking sequence for receiving and shipping trucks and their assignment to receiving and shipping docks are expected results of this research as well as product routing from receiving to shipping trucks.

1.4 Organization of the thesis
Thesis is divided into seven chapters. Chapter one, Introduction, describes basics of cross docking and its role within logistics. Furthermore, research objectives of the thesis are stated. Chapter two, Literature review, is a review of the research conducted in the area of cross docking with the main focus on optimizing different operations of cross docking. Chapter three, Model description, describes the particular cross docking model studied including assumptions made for this model and model development. Chapter four, Mathematical model, presents the mathematical model solution developed to solve the cross docking model. Chapter five, Heuristic method, presents the heuristic algorithm developed to solve the cross docking model including step by step algorithms and flow charts. Chapter six, Results and conclusions, focuses firstly on results of mathematical model and heuristic algorithm applied for data sets provided in Appendix B. and secondly, it summarizes the results of the research achieved in the previous chapters. After conclusion, list of references and three appendixes follow.
2 Literature review
Overall, there are not so many research works available which are concerned with cross docking operation and improvement of its processes. Moreover, each research is focused on a particular process of cross docking. In this chapter, some of these studies are discussed, particularly the ones which concern shape of cross dock facility, dock door assignment, positioning of products in temporary storages, and truck scheduling. Results of these research studies were taken into account in order to improve the outcomes of this thesis.

2.1 Shape of cross dock facility
Cross dock facility design is an important part of implementing cross docking because it can to a great extent influence the efficiency of the facility and costs connected with its operation. Shape of the cross dock depends on the size of the facility. Common shapes for a dock are I, L, T even though it is also possible to find unusual ones as well such as U, H or E.

Among others, Bartholdi and Gue (2004) were concerned with finding the best shape for a cross dock facility. Within their study, they focus on cross dock design and when deciding for the best shape, they consider issues such as determining the number of receiving and shipping doors, minimum width of dock, and possibility of later increase of the facility. They are intentionally leaving aside issues which can and in reality do compromise the cross dock design such as parking requirements, turning radius of trucks or the need for office or maintenance building as well as the fact, that not all companies which want to implement cross docking have the possibility to build a new facility but they are more likely going to use a building formerly designed for another purpose.
To determine which shape is the best under specified conditions, they carried out experiments with several shapes (I, L, T, H) and estimate of labor cost based on average travel distance. For each shape, they explored four characteristics – size (40 to 350 doors), layout, patterns of freight flows and fraction of doors devoted to receiving.
The results of their study shows that with respect to the labor cost, the best shape for small to mid-sized cross docks is a narrow rectangle or I-shape and for big-sized facilities (above 200 doors) X-shape is the best; shapes like L or U should be avoided because of the costs of the additional corners.

2.2 Dock door assignment problem
One of the key problems of cross docking is the dock door assignment problem, which involves assigning particular doors for incoming and leaving trucks so that the travel distance of the products within the cross dock facility is minimized. There exist more studies which are concerned with dock door assignment problem; however, each of them is offering its own solutions and methodologies without actually resulting in one approved method how to solve the dock door assignment problem.
In their study, Aickelin and Adewunmi (2006) proposed a two-step solution which consists of (1) simulation of cross dock door assignment problem with the aim to get
better understanding of the doors assignment, and (2) optimizing the results of simulation models. The simulation is supposed to imitate the cross dock facility performance in a controlled environment in order to estimate the real performance, taking into account parameter changes like cross dock facility shape, the number of receiving and shipping doors, trailer schedule, freight flow and mix over period in time. The optimization of the simulation models is then considered as a search for an optimal solution for these models with as many decision variables as needed. The methods they use for their modeling are genetic algorithm for the simulation and memetic algorithm for the optimization part. Result of there modeling is proposition of a new methodology for solving door dock assignment problem.

2.3 Positioning of goods in temporary storage

One of the basic principles of cross docking is that products arriving to the facility are being shipped within maximum of 24 hours since their arrival and so one of the advantages of cross docking is that the warehouse space is limited and all the goods is being shipped as soon as possible. However, it is necessary to have temporary storage(s) within the cross dock facility for incoming goods which is waiting for being shipped. Finding the most suitable location for temporary storage is another big issue in cross docking related research.

To determine temporary storage locations for incoming goods so that the travel distances of these products within the facility are minimized, is the goal of the study by Vis and Roodbergen (2008). They developed a method consisting of a network formulation which includes loading and unloading dock door locations, travel distances and available storage space in the facility. The method can be adapted so it enables applications for situations with different layouts and for situation with higher time pressure in loading than unloading. Various experiments conducted showed that it is possible to use the method both in the operational and design phases. Also, compared to the commonly used heuristic method, this method shows up to about 40% difference.

2.4 Truck scheduling

Another important issue in the area of cross dock is the scheduling within the facility. The reason is that one way to reduce costs in a cross dock is to arrange the docking of the incoming and outgoing trucks in such a way that loads can be efficiently moved across the facility. By efficiently it is meant that the distances between corresponding incoming and outgoing truck are the shortest possible, which requires door assignment based on the destination of incoming products.

There exist many ways how to find the optimal scheduling of trucks in cross dock, Ley and Elfayoumy (2007) used for this purpose genetic algorithm. They used genetic algorithm to create a truck schedule so that the time a truck must spend unloading and uploading at a cross dock facility is decreased. Furthermore, the time and accuracy evaluation testing proved that using genetic algorithm in case of small problem sizes (4, 5, 6 and 7 incoming trucks) provides an accurate and timely solution. However as they
admit, using genetic algorithm is only one way to do the scheduling in a cross dock facility, and not necessarily the best one.

The most important source concerned with both this particular topic and the whole thesis is a dissertation by Yu (2002). Yu in his work generated various cross docking models based on the number of docks available at the facility, the dock holding pattern for trucks, and the existence of temporary storage. Dock holding pattern is defined as “stay” or “repeat”, where “stay” means that either receiving or shipping truck once it enters the dock it has to stay until its task is finished (unloaded or loaded); on the other hand “repeat” means that either receiving or shipping truck can enter or leave the dock repeatedly independently on finishing its task. Based on possible combinations of these three criteria (number of docks, dock holding pattern, existence of temporary storage) as shown in figure 2-1, Yu generated thirty-two possible cross docking models.

Out of these thirty-two models generated, the author modeled and solved three of them as presented in table 2-1.

This thesis focuses on the following model as shown in table 2-2 using the same framework as Yu’s models.
Table 2-2. Model studied in this thesis

<table>
<thead>
<tr>
<th>Receiving area</th>
<th>Temporary storage</th>
<th>Shipping area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of docks</td>
<td>Dock holding pattern</td>
<td>Number of docks</td>
</tr>
<tr>
<td>Multiple</td>
<td>Stay</td>
<td>Yes</td>
</tr>
</tbody>
</table>

In further research based on his dissertation (Yu, 2002), Yu and Egbelu (2008) aimed to find the best incoming and outgoing trucks scheduling sequence to minimize the makespan (total operation time) when the temporary storage is placed at the shipping dock, and the product assignments to both incoming and outgoing trucks are determined simultaneously. To solve this problem, authors used three different solution approaches. Within the first approach, a mathematical model was developed with the goal to minimize the makespan of a cross docking operation. The second approach uses complete enumeration to generate all possible truck scheduling sequences. To increase solution efficiency, heuristic algorithm is employed as a part of the third approach. The main goal of the heuristic algorithm is to minimize the total number of products needed to be stored in the temporary storage. In the first stage of this algorithm, the best subset of associate inbound trucks that are the best candidates to transfer loads to each unscheduled outbound truck is found; in the second phase, one of the unscheduled outbound truck and its associate inbound trucks are selected and scheduled. In general, the compound heuristic algorithm resulted in solutions which were very close to the global optimal solutions.

### 2.5 Miscellaneous

In addition to the above mentioned four main areas of research, there exist some other studies concerned with more general or more specialized problems connected to cross-docking, including numerous case studies describing cross-docking facilities in particular companies.

One of issues studied is the location of cross-dock facilities and distribution centers within the supply chain networks, and the role of cross-docking within supply chain management. Ross and Jayaraman (2008) were experimenting with a new heuristics solution procedures for the location of cross-docks and distribution centres in supply chain management design; the conclusion of their work is that for this kind of tasks, integrating traditional simulated annealing with TABU search is recommended.

The problem of the shape of cross-dock facility discussed above is closely connected to the design of the whole warehouse, particularly the placement of the inventory and inventory routes within the warehouse/cross-dock facility (inventory management). Baker and Canessa (2009) sum up current literature on the methodology of warehouse design including issues like steps to be taken when designing a warehouse, individual tools and techniques used within each step etc., and concluding that a more
comprehensive and systematic methodology is needed. Heragu et al. (2005) were concerned with product allocation and determination of size of functional area in a warehouse design; in their study, they used mathematical model and a heuristic algorithm to solve these two problems in order to minimize annual handing and storage costs.

Lately, research is also concerned with the importance of effective real-time information flow among all parties using cross-docking. Study by Babics (2005) describes the advantages and drawbacks of joining third party logistics provider as well as challenges connected with information flows and communication within all the parties. Mason et al (2003) are writing about global inventory visibility as a result of integrating warehouse management system (information on supplier/customer warehouse inventory levels and key customer ordering patterns) and transportation management systems (products and vehicles information).

Various case studies build a quite big part of the research; they are mainly devoted to the situation of cross dock implementation or functioning in different companies, and to analysis and optimization of the current processes. The aim of the case studies apart from their descriptive function is also analysis of current processes followed by various optimization methods. For example, Bartholdi & Gue (2000) devoted their work to reducing labor costs of transferring freight at less-than-truckload cross-docking terminals; the results of their research, based on modeling of travel costs and three different types of congestion, were successfully implemented at the terminal in Stockton, California. Another example of case study approach is report Cross-docking speeds up Belk’s distribution center (2005) about Belk, a U.S. department store chain; the efficiency of goods supply from vendors to the customers was very low and the costs were too high, which exposed the company to higher competitive risks. The changes which were made by the company management included building of a brand new highly automated, cross-docked distribution center located relatively near to all the 215 stores - this brought among others faster supplies, freight costs saving and minimizing of merchandise handling.

For this literature review, four different processes within cross docking were selected (shape of cross dock facility, dock door assignment, positioning of goods in temporary storage, and truck scheduling) with the aim to present results of research conducted in these particular ideas and also, to use the results of these studies for improving the outcomes of this thesis. In addition, few other studies concerned with various areas of cross-docking were shortly presented.
3 Model description

The cross docking model studied in this research is one of the thirty-two models identified in Yu (2002); the description of how the models were generated is presented in section 2.4. The definition of the makespan for this model is also adopted from Yu’s research. Moreover, makespan is defined as the total operating time of the cross docking operation. The total operating time is from the moment when the first receiving truck enters a receiving dock to the moment the last shipping truck leaves a shipping dock. Furthermore, it should be mentioned that in this thesis other operations of cross docking system such as sorting, labeling, etc are not considered.

In a basic cross docking model within this study, receiving trucks arrive at receiving docks and unload all initially loaded products to receiving docks. Then products move from receiving docks using movement system to shipping docks where shipping trucks load the products and leave the cross dock facility.

Figure 3-1 is a presentation of the model used and shows the product routing inside the facility. Products can be transferred from any receiving dock to any shipping dock(s) as well as to the temporary storage.

In this model, it is assumed that there are multiple numbers of receiving and shipping docks which are not necessarily equal. In addition, receiving and shipping trucks can enter receiving and shipping docks only once and they can not leave docks before they finish their tasks which is for receiving trucks to unload all initially loaded products and for shipping trucks to load all initially needed products. Furthermore, it is also assumed that there is a temporary storage in the cross docking system in which products unloaded from receiving trucks not needed by docking shipping trucks can stay for a short time. Capacity of the temporary storage is assumed to be unlimited which means all products loaded in all receiving trucks can be transferred to the temporary storage without any problem.

Figure 3-1. Product routing in the cross docking model
3.1 Assumptions

Following assumptions are applied to this model:

1. Cross dock facility is capable of receiving and shipping both receiving and shipping trucks at the same time. Receiving and shipping trucks can enter different receiving and shipping docks simultaneously.
2. All receiving and shipping trucks are available at time zero. Receiving and shipping trucks are not differentiated by availability time.
3. All received products have to be shipped without being stored for a long period of time (up to 24 hours). Total number of units of products initially loaded in all receiving trucks is equal to total number of units of products initially needed by all shipping trucks.
4. There is no order for unloading or loading products. This means that any product type can be unloaded from a receiving truck or a temporary storage regardless of its location in the receiving truck or temporary storage. For example, if a receiving truck is loaded as shown in picture below, products type B can be unloaded before products types C, A or D.

```
  C  A  D  B
```

5. Products are interchangeable. It means that one type of product can be transferred to any shipping truck which needs that particular type of product regardless of its destination. For example, products type A unloaded from a receiving truck can be transferred to one or more shipping trucks which need products type A.
6. It is possible to unload only needed amount of products from a receiving truck. This means that loaded products in a receiving truck are accessible and that any portion of each product type can be unloaded as needed. For example, suppose that a receiving truck is loaded with 200 units of product type A and 100 units of product type C. If 50 units of load type A and 50 units of load type C are needed at the moment, they can be unloaded first.
7. Receiving trucks do not have to unload their products into receiving docks immediately. Any portion of products can stay in receiving trucks as long as receiving trucks are at the receiving docks.
8. There is no time limit for docking receiving and shipping trucks. Receiving and shipping trucks can stay at receiving and shipping docks for unlimited time before they finish their tasks. In addition, they leave immediately when they are finished.
9. Only one unit of product can be loaded into a shipping truck at a time. Products from different receiving trucks or from the temporary storage can not be loaded into the shipping truck simultaneously.
10. Other operations of cross dock center such as scanning and sorting are not considered. Therefore, it is assumed that receiving products can be loaded as they arrive in a receiving dock.
11. There is a fixed interval between products traveling on movement system.
12. Loading time of products is equal to their unloading time.
13. Each product type has its own loading or unloading time.
14. Travel times for products from different receiving docks to different shipping docks are not equal. Also travel times from different receiving docks to the temporary storage are not equal. Similarly, travel times from the temporary storage to different shipping docks are also not equal.
15. There is only one temporary storage in the facility and the capacity of this temporary storage is unlimited.
16. Delay time for truck changes is the same for all receiving and shipping trucks.
17. The following information are assumed to be known:
   i. Number of receiving docks.
   ii. Number of shipping docks.
   iii. Number of receiving trucks.
   iv. Number of shipping trucks.
   v. Product types and number of units of products loaded in a receiving truck.
   vi. Product types and number of units of products needed for a shipping truck.
   vii. Loading or unloading time for each product type.
   viii. Delay time for truck change.
   ix. Distances between receiving docks and shipping docks.
   x. Distances between receiving docks and the temporary storage as well as distances between the temporary storage and shipping docks.

3.2 Expected results
Solution of this model is expected to provide the following results:
1. Total operation time for cross docking operation.
2. Docking sequence of the receiving trucks at receiving docks.
3. Docking sequence of the shipping trucks at shipping docks.
4. Receiving and shipping trucks docking and departure times.
5. Product routing from receiving trucks to shipping trucks including products passing through the temporary storage.
Solutions developed for this model are presented in chapters 4 and 5 and the results are presented in chapter 6.

3.3 Model development
In this research, two different methods were used to solve the cross docking problem, mathematical modeling and heuristic algorithm. For the first method, a mixed integer programming model was developed to minimize total operation time. This model was formulated in AMPL modeling language and used to solve small size problems because of the limitations of student version of AMPL and the computational time required by this method. For the second method, a heuristic algorithm was developed to find solutions fast and was used for problems of larger size. In order to examine the performance of heuristic algorithm, small problems were solved by both mathematical
model and the heuristic algorithm. The mathematical model developed for this problem is presented in chapter 4 and the heuristic method is presented in chapter 5. In addition, results and comparison of these methods are presented in chapter 6.
4 Mathematical model

For the mathematical model of the problem, assumptions from section 3.1 were made. In addition, it is assumed that the number of receiving and shipping docks in the model cannot be higher than number of receiving and shipping trucks.

Following notations were used in the mixed integer programming model.

4.1 Notations

\( T \) = Makespan,

\( d_{im}^r \) = Time at which receiving truck \( i \) enters receiving dock \( m \),

\( l_{im}^r \) = Time at which receiving truck \( i \) leaves receiving dock \( m \),

\( d_{jn}^s \) = Time at which shipping truck \( j \) enters shipping dock \( n \),

\( l_{jn}^s \) = Time at which shipping truck \( j \) leaves shipping dock \( n \).

Integer variables

\( x_{ijk} \) = Number of units of product type \( k \) which transfer from receiving truck \( i \) to shipping truck \( j \).

Binary variables

\( t_{ij} \) = \begin{cases} 1, & \text{If any products transfer from receiving truck } i \text{ to shipping truck } j \\ 0, & \text{Otherwise} \end{cases}

\( p_{ij} \) = \begin{cases} 1, & \text{If receiving truck } i \text{ preceeds receiving truck } j \text{ in the receiving truck sequence} \\ 0, & \text{Otherwise} \end{cases}

\( q_{ij} \) = \begin{cases} 1, & \text{If shipping truck } i \text{ preceeds shipping truck } j \text{ in the shipping truck sequence} \\ 0, & \text{Otherwise} \end{cases}

\( A_{im}^r \) = \begin{cases} 1, & \text{If receiving truck } i \text{ is assigned to receiving dock } m \\ 0, & \text{Otherwise} \end{cases}

\( A_{jn}^s \) = \begin{cases} 1, & \text{If shipping truck } j \text{ is assigned to shipping dock } n \\ 0, & \text{Otherwise} \end{cases}

Data

\( R \) = Number of receiving trucks,

\( S \) = Number of shipping trucks,

\( M \) = Number of receiving docks,

\( N \) = Number of shipping docks,

\( P \) = Number of product types,
\[ p_{ik}^r = \text{Number of units of product type } k \text{ that is initially loaded in receiving truck } i, \]
\[ p_{jk}^s = \text{Number of units of product type } k \text{ that is initially needed for shipping truck } j, \]
\[ h_k = \text{Handling time for one unit of product type } k \text{ (loading or unloading time),} \]
\[ W_{mn} = \text{Moving time from receiving dock } m \text{ to shipping dock } n \text{ (distance between doors),} \]
\[ D = \text{Delay time for truck change,} \]
\[ Q = \text{Big number.} \]
4.2 Mixed integer programming model

The mixed integer programming model for the scheduling problem is presented as:

Min $T$

Subject to

\[ T \geq l_{jn}^i, \quad \text{for all } j, n \]  \hspace{1cm} (4-1)

\[ \sum_{m=1}^{M} A_{im}^i = 1, \quad \text{for all } i \]  \hspace{1cm} (4-2)

\[ \sum_{i=1}^{R} A_{im}^i \geq 1, \quad \text{for all } m \]  \hspace{1cm} (4-3)

\[ \sum_{n=1}^{N} A_{jn}^i = 1, \quad \text{for all } j \]  \hspace{1cm} (4-4)

\[ \sum_{j=1}^{S} A_{jn}^i \geq 1, \quad \text{for all } n \]  \hspace{1cm} (4-5)

\[ \sum_{i=1}^{R} x_{ijk} = p_{jk}^i, \quad \text{for all } j, k \]  \hspace{1cm} (4-6)

\[ \sum_{j=1}^{S} x_{ijk} = p_{ik}^j, \quad \text{for all } i, k \]  \hspace{1cm} (4-7)

\[ x_{ijk} \leq Q_{ij}, \quad \text{for all } i, j, k \]  \hspace{1cm} (4-8)

\[ l_{im}^i \geq d_{im}^i + A_{im}^i \sum_{h=1}^{P} p_{ik}^h h_k, \quad \text{for all } i, m \]  \hspace{1cm} (4-9)
\[ d'_{jm} \geq l'_{im} + D - Q(1 - p_{ij}), \quad \text{for all } i, j, m \text{ where } i \neq j \quad (4-10) \]

\[ d'_{im} \geq l'_{jm} + D - Qp_{ij}, \quad \text{for all } i, j, m \text{ where } i \neq j \quad (4-11) \]

\[ d'_{im} \geq d'_{jm} - Qp_{ij} - Q(1 - A'_{im}) - Q(1 - A'_{jm}), \quad \text{for all } i, j, m, n \text{ where } i \neq j, m \neq n \quad (4-12) \]

\[ p_{ii} = 0, \quad \text{for all } i \quad (4-13) \]

\[ l'_{jn} \geq d'_{jn} + A'_{jn} \sum_{k=1}^{p} p'_{jk} h_k, \quad \text{for all } j, n \quad (4-14) \]

\[ d'_{jn} \geq l'_{in} + D - Q(1 - q_{ij}), \quad \text{for all } i, j, n \text{ where } i \neq j \quad (4-15) \]

\[ d'_{in} \geq l'_{jn} + D - Qq_{ij}, \quad \text{for all } i, j, n \text{ where } i \neq j \quad (4-16) \]

\[ d'_{im} \geq d'_{jn} - Qq_{ij} - Q(1 - A'_{im}) - Q(1 - A'_{jn}), \quad \text{for all } i, j, m, n \text{ where } i \neq j, m \neq n \quad (4-17) \]

\[ q_{ii} = 0, \quad \text{for all } i \quad (4-18) \]

\[ l'_{jn} + Q(1 - A'_{jn}) \geq d'_{im} + W_{mn} + \sum_{k=1}^{p} x_{ijk} h_k - Q(1 - t_{ij}) - Q(1 - A'_{im}), \quad \text{for all } i, j, m, n \quad (4-19) \]

\[ l'_{jn} + Q(1 - A'_{jn}) \geq l'_{im} - Q(1 - t_{ij}) - Q(1 - A'_{im}), \quad \text{for all } i, j, m, n \quad (4-20) \]

\[ \text{all variables } \geq 0 \quad (4-21) \]
Constraint (4-1) makes makespan equal to the time at which the last shipping truck leaves a shipping dock. Constraint (4-2) makes sure that each receiving truck is assigned to only one receiving dock. Constraint (4-3) assigns each receiving dock to at least one receiving truck in order to utilize all receiving docks. This constraint is valid for the case that number of receiving docks is not higher than the number of receiving trucks which is a part of assumptions for the mathematical model. Similarly, constraints (4-4) and (4-5) control the assignment of shipping trucks to shipping docks. Constraint (4-6) sets the relation between products transferring from all receiving trucks to each shipping truck. In addition, this constraint makes the sum of all products transferring from all receiving trucks to a shipping truck equal to the number of products initially needed by that shipping truck. Similarly, constraint (4-7) sets the relation between products transferring from each receiving truck to all shipping trucks. This is done by setting the total number of products transferring from each receiving truck to all shipping trucks equal to the total number of products which are initially loaded in each receiving truck. Constraint (4-8) sets the relation between product transfer variable and the decision variable $t$. Constraint (4-9) sets the time at which receiving truck $i$ leaves receiving dock $m$ to be greater or equal to the time at which receiving truck $i$ enters receiving dock $m$ plus the time it takes to unload all its products. This equation is effective only if receiving truck $i$ is assigned to receiving dock $m$. Constraints (4-10) and (4-11) adjust entering and leaving time for different receiving trucks to a same receiving dock based on their order in the receiving truck sequence. Constraint (4-12) adjusts the entering time of different receiving trucks to different receiving docks based on their order in the receiving truck sequence. Constraint (4-13) ensures that a receiving truck does not precede itself in the receiving truck sequence. Constraints (4-14) sets the time at which shipping truck $j$ leaves shipping dock $n$ to be greater or equal to the time at which shipping truck $j$ enters shipping dock $n$ plus the time it takes to load all its needed products. This equation is effective only if shipping truck $j$ is assigned to shipping dock $n$. Constraints (4-15) and (4-16) adjust entering and leaving time for different shipping trucks to the same shipping dock based on their order in the shipping truck sequence. Constraint (4-17) adjusts the entering time of different shipping trucks to different shipping docks based on their order in the shipping truck sequence. Similarly, constraint (4-18) ensures that a shipping truck does not precede itself in the shipping truck sequence. Constraint (4-19) makes a relation between leaving time of a shipping truck and entering time of a receiving truck if there is any product to be transferred from that receiving truck to the shipping truck. Furthermore, this constraint has a role in dock assignment based on the distance of the receiving dock to the shipping dock which these trucks are assigned to. Constraint (4-20) ensures that no shipping truck will leave the shipping dock before the receiving truck which is receiving products from. In addition, as it is possible that more than one shipping truck is receiving products from one receiving truck at the same time, this constraint balances the timing for the loading shipping trucks.

In this mixed integer programming model, number of decision variables and constraints are presented below.

Total number of variables = $R^2 + S^2 + RS(P + 1) + 3(RM + SN)$
Number of integer variables = $RSP$
Number of binary variables = $R(R+S+M)+S(S+N)$
Number of continuous variables = 2(RM+SN)
Total number of constraints =
\[ R^2(M + M^2) + S^2(N + N^2) + 2(R + S + RSMN) + M(1 - RM) + N(1 - SN) + P(R + S + RS) \]

In this model, value of decision variable \( x_{ijk} \) shows product routing from receiving trucks to shipping trucks. In addition, value of decision variables \( p_{ij} \) and \( q_{ij} \) shows receiving and shipping truck docking sequence respectively. For example, it is assumed that the values of \( p_{ij} \) and \( q_{ij} \) after solving a problem are as shown in table 4-1.

<table>
<thead>
<tr>
<th></th>
<th>p</th>
<th></th>
<th></th>
<th>q</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>j</td>
<td></td>
<td></td>
<td>i</td>
<td>j</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
<td></td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0</td>
<td></td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td></td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td></td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td></td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
<td></td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td></td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

According to table 4-1, docking sequence for receiving trucks is 2, 3, 1 and docking sequence for shipping trucks is 1, 3, 2. When the value of \( p_{ij} \) or \( q_{ij} \) is 1, it means that receiving or shipping truck \( i \) precedes receiving or shipping truck \( j \) in receiving or shipping truck docking sequence respectively. Similarly, receiving and shipping docks assignment can be found based on the values of \( A'_{im} \) and \( A'_{jn} \) resulted in the model.

### 4.3 Implementation

AMPL modeling language is used to formulate the mixed integer programming model introduced in previous section. In addition, CPLEX 11.2.0 solver is used as well to solve the model. Formulated model is presented in appendix A.

The model was used to solve different test data sets which are presented in appendix B and the results are presented in chapter 6. Because of the limitations of the student version of AMPL, only very small data sets were used for the mathematical model. In addition to this limitation, mathematical model takes more time and computational power as the data sets grow bigger. Therefore, heuristic method was used to solve the problem and to solve larger data sets. The heuristic algorithm developed for this problem is presented in chapter 5.
5 Heuristic method

In developing the heuristic algorithm for the scheduling problem, the same assumptions were made as for the mathematical model with one exception. In the heuristic algorithm, number of receiving or shipping docks can be higher than the number of receiving or shipping trucks. The main strategy to select receiving and shipping trucks to be scheduled is the number of products that can be transferred from a receiving truck to a shipping truck directly. The heuristic algorithm for the scheduling problem minimizes the number of product transfer from receiving trucks to the temporary storage by using all available receiving and shipping docks. The heuristic algorithm is explained in detail in section 5.1.2, and the step by step algorithm is presented in section 5.2.

5.1 Notations

For the heuristic algorithm of the scheduling problem, following notations are used:

Data:

\[ R = \text{Number of receiving trucks in the set,} \]
\[ S = \text{Number of shipping trucks in the set,} \]
\[ M = \text{Number of receiving docks in the set,} \]
\[ N = \text{Number of shipping docks in the set,} \]
\[ P = \text{Number of product types in the set,} \]
\[ p_{r_{ik}} = \text{Number of units of product type } k \text{ that is initially loaded in receiving truck } i, \]
\[ p_{s_{jk}} = \text{Number of units of product type } k \text{ that is initially needed for shipping truck } j, \]
\[ h_k = \text{Handling time for one unit of product type } k \text{ (loading or unloading time),} \]
\[ W_{mn} = \text{Travel time from receiving dock } m \text{ to shipping dock } n \text{ (distance between docks),} \]
\[ W_{ms} = \text{Travel time from receiving dock } m \text{ to the temporary storage (distance between receiving docks and the temporary storage),} \]
\[ W_{sn} = \text{Travel time from the temporary storage to shipping dock } n \text{ (distance between the temporary storage and shipping docks),} \]
\[ D = \text{Delay time for truck change.} \]

Variables:

\[ r_{\text{last}} = \text{Last selected receiving truck,} \]
\[ s_{\text{last}} = \text{Last selected shipping truck,} \]
\[ r^d_i = \text{Receiving dock in which the receiving truck } i \text{ is docking,} \]
\[ s^d_j = \text{Shipping dock in which the shipping truck } j \text{ is docking.} \]
Time:
\[ d^r_{im} = \text{Time at which receiving truck } i \text{ enters the receiving dock } m, \]
\[ l^r_{im} = \text{Time at which receiving truck } i \text{ leaves the receiving dock } m, \]
\[ d^s_{jn} = \text{Time at which shipping truck } j \text{ enters the shipping dock } n, \]
\[ l^s_{jn} = \text{Time at which shipping truck } j \text{ leaves the shipping dock } n, \]
\[ l^r_{md} = \text{Time at which the last receiving truck docking at receiving dock } m \text{ leaves it. In other words, the time at which the receiving dock } m \text{ becomes available}, \]
\[ l^s_{nd} = \text{Time at which the last shipping truck docking at shipping dock } n \text{ leaves it. In other words, the time at which the shipping dock } n \text{ becomes available}, \]
\[ l^{storage} = \text{Time at which the temporary storage can receive or send products.} \]

Number of products:
\[ p^r_{ik} = \text{Number of units of product type } k \text{ that is currently loaded in receiving truck } i, \]
\[ p^s_{jk} = \text{Number of units of product type } k \text{ that is currently needed for shipping truck } j, \]
\[ p^{rst}_{k} = \text{Number of units of product type } k \text{ stored in the temporary storage}, \]
\[ T^{temp}_{ijk} = \text{Number of units of product type } k \text{ which can be transferred from receiving truck } i \text{ to shipping truck } j, \]
\[ T_{ijk} = \text{Number of units of product type } k \text{ which transfers from receiving truck } i \text{ to shipping truck } j, \]
\[ T^{toST}_{ik} = \text{Number of units of product type } k \text{ which transfers from receiving truck } i \text{ to the temporary storage}, \]
\[ T^{fromST}_{jk} = \text{Number of units of product type } k \text{ which transfers from the temporary storage to shipping truck } j. \]

Truck:
\[ r_i = \text{Receiving truck } i, \]
\[ s_j = \text{Shipping truck } j. \]

Dock:
\[ d^r_m = \text{Receiving dock } m, \]
\[ d^s_n = \text{Shipping dock } n. \]

Set:
\[ R^{docking} = \text{Set of receiving trucks which are currently docking in the receiving docks}, \]
\[ S^{docking} = \text{Set of shipping trucks which are currently docking in the shipping docks}, \]
\[ R_{\text{scheduled}} = \text{Set of scheduled receiving trucks}, \]
\[ S_{\text{scheduled}} = \text{Set of scheduled shipping trucks}, \]
\[ R_{\text{remaining}} = \text{Set of receiving trucks which still have some loads to unload. This includes unscheduled receiving trucks and receiving trucks which are in receiving docks and are not finished yet.} \]
\[ S_{\text{remaining}} = \text{Set of shipping trucks which still need some loads to load. This includes unscheduled shipping trucks and shipping trucks which are in shipping docks and are not finished yet.} \]
\[ A^{rd} = \text{Set of available receiving docks}, \]
\[ A^{sd} = \text{Set of available shipping docks}. \]

### 5.2 Heuristic algorithm of the scheduling problem

The heuristic algorithm developed for the scheduling problem is a continuous process of searching for receiving and shipping trucks to schedule, based on a simple strategy. This strategy is to find receiving and shipping trucks with the highest number of product transfer from receiving to shipping trucks. The algorithm always works with receiving truck \( r^{\text{last}} \) called the last selected receiving truck \( r^{\text{last}} \) and shipping truck \( s^{\text{last}} \) called the last selected shipping truck \( s^{\text{last}} \). These so called last selected receiving and shipping trucks are selected based on different situations of receiving and shipping trucks as well as on the situation of receiving and shipping docks in each iteration of the algorithm.

In the beginning, the algorithm calculates the number of units of products that can be transferred from a receiving truck to every shipping truck. This is done for all receiving and shipping trucks in the set and it is called temporary product transfer \( \sum_{k=1}^{p} T_{ijk}^{\text{temp}} \) from receiving truck \( i \) to shipping truck \( j \). Every time when there is a real product transfer in the algorithm, \( \sum_{k=1}^{p} T_{ijk}^{\text{temp}} \) is updated for all receiving and shipping trucks using procedure 1 (Updating temporary product transfer procedure) which is presented in section 5.2.4.

After calculating \( \sum_{k=1}^{p} T_{ijk}^{\text{temp}} \) for all receiving and shipping trucks, the algorithm finds a pair of receiving and shipping trucks with the highest number of product transfer to be scheduled as the first ones using procedure 2 (Finding a pair of unscheduled receiving and shipping trucks to schedule) presented in section 5.2.5. In this point, after the receiving truck \( r^{\text{last}} \) and the shipping truck \( s^{\text{last}} \) are selected, the algorithm assigns them to appropriate receiving and shipping dock respectively using procedure 6 (Receiving dock assignment procedure) and procedure 7 (Shipping dock assignment procedure) presented in sections 5.2.9 and 5.2.10 respectively. Furthermore, the algorithm virtually transfers the products from receiving truck \( j^{\text{last}} \) to the shipping truck \( s^{\text{last}} \) and updates timing for the receiving and the shipping truck. Timing of receiving and shipping trucks is discussed in section 5.2.2 and the timing procedure (procedure 8) is presented in section 5.2.11.
After each product transfer, the algorithm updates the number of units of products remaining loaded \((\sum_{k=1}^{p} p_{rk}^{'r,k})\) in receiving truck \(r_{last}\) and the number of units of products still needed \((\sum_{k=1}^{p} p_{sk}^{'s,k})\) by shipping truck \(s_{last}\) as well as \(\sum_{k=1}^{p} T_{ijk}^{temp}\) (temporary product transfer) for all receiving and shipping trucks.

Based on the number of units of products remaining in receiving truck \(r_{last}\) and the number of units of products needed by shipping truck \(s_{last}\), there are four possible scenarios; each of them uses different approach to select next receiving or shipping truck. These four possible scenarios are:

1) Both the receiving and the shipping truck are satisfied.
2) The receiving truck is satisfied but the shipping truck is not satisfied.
3) The receiving truck is not satisfied but the shipping truck is satisfied.
4) Neither the receiving nor the shipping truck is satisfied.

Descriptions of these four scenarios and corresponding actions taken in each case are:

1) In case that both the receiving and the shipping truck are satisfied \((\sum_{k=1}^{p} p_{rk}^{'r,k} = 0 \text{ and } \sum_{k=1}^{p} p_{sk}^{'s,k} = 0)\), both the receiving and the shipping truck leave the receiving and the shipping docks. In addition, the receiving truck will be removed from the list of remaining receiving trucks \((R_{remaining} = R_{remaining} - \{r_{last}\})\). Similarly, the shipping truck will be removed from the list of remaining shipping trucks \((S_{remaining} = S_{remaining} - \{s_{last}\})\) and the algorithm finds another pair of receiving and shipping truck with the highest number of product transfer from receiving to shipping truck using procedure 2 presented in section 5.2.5.

2) In case that the receiving truck is satisfied but the shipping truck is not satisfied \((\sum_{k=1}^{p} p_{rk}^{'r,k} = 0 \text{ and } \sum_{k=1}^{p} p_{sk}^{'s,k} > 0)\), the receiving truck leaves the receiving dock and will be removed from the remaining receiving truck list \((R_{remaining} = R_{remaining} - \{r_{last}\})\) and the shipping truck will stay in the shipping dock. Then the algorithm finds an unscheduled receiving truck with the highest number of product transfer to the shipping truck \(s_{last}\) using procedure 3 (Finding an unscheduled receiving truck to schedule) presented in section 5.2.6.

3) Similarly, if the receiving truck is not satisfied but the shipping truck is satisfied \((\sum_{k=1}^{p} p_{rk}^{'r,k} > 0 \text{ and } \sum_{k=1}^{p} p_{sk}^{'s,k} = 0)\), then the receiving truck stays in the receiving dock.
dock and the shipping truck leaves the shipping dock and is removed from the list of remaining shipping trucks ($S_{\text{remaining}} = S_{\text{remaining}} \setminus \{ s_{\text{last}} \}$). The algorithm finds an unscheduled shipping truck with the highest number of units of products receiving from the receiving truck $r_{\text{last}}$ using procedure 4 (Finding an unscheduled shipping truck to schedule) presented in section 5.2.7.

4) In case that neither the receiving nor the shipping truck is satisfied ($\sum_{k=1}^{P} p_{r_{\text{last}k}} > 0$ and $\sum_{k=1}^{P} p_{s_{\text{last}k}} > 0$), the algorithm takes an action based on the availability of receiving and shipping docks. Therefore, in case that there are both receiving and shipping docks available ($| A^{rd} | > 0$ and $| A^{sd} | > 0$), the last selected receiving and the last selected shipping truck will stay in the docks and the algorithm finds another pair of receiving and shipping trucks to schedule in a similar way as the first scenario using procedure 2 presented in section 5.2.5. The advantage of scheduling another pair is that there will be no conflict in product transfer from receiving trucks to the shipping trucks and they will transfer their products simultaneously. In case that there is a receiving dock available but no shipping dock is available ($| A^{rd} | > 0$ and $| A^{sd} | = 0$), the algorithm finds another unscheduled receiving truck with highest number of product transfer to shipping truck $s_{\text{last}}$ similar to second scenario using procedure 3 presented in section 5.2.6. Similarly, in case that there is not a receiving dock available but there is a shipping dock available ($| A^{rd} | = 0$ and $| A^{sd} | > 0$), the algorithm finds an unscheduled shipping truck with the highest number of product transfer from receiving truck $r_{\text{last}}$ using procedure 4 presented in section 5.2.7. Furthermore, if there will be neither receiving nor shipping dock available ($| A^{rd} | = 0$ and $| A^{sd} | = 0$), then the algorithm tries to find any product transfer from docking receiving trucks and the temporary storage to docking shipping trucks using procedure 5 (Finding product transfer between docking trucks). If the algorithm finds any products to be transferred, it transfers the products to shipping trucks and as a result if any of docking receiving or docking shipping trucks is satisfied, then algorithm starts over by finding another truck to schedule. Otherwise, the algorithm finds the docking receiving truck with minimum number of products remaining loaded and transfers the products from the receiving truck to the temporary storage. Consequently, the receiving truck leaves the receiving dock and algorithm can schedule another unscheduled receiving truck. Product transfer procedure (procedure 9) is presented in section 5.2.12.

The algorithm continues scheduling receiving and shipping trucks until there is no receiving and no shipping truck left unscheduled ($| R_{\text{remaining}} | = 0$ and $| S_{\text{remaining}} | = 0$) and then calculates the total operation time. The operation time is the time at which the
last shipping truck leaves the shipping dock minus the time at which the first receiving truck enters the receiving dock and is calculated.
The scheduling algorithm is presented in section 5.2.3 and also coded in visual basic programming language in appendix C.

5.2.1 Receiving and shipping dock assignment
In this algorithm, distances between receiving and shipping docks are taken into account. Every time when a new receiving or shipping truck is scheduled, procedures for dock assignment are used to assign the receiving or the shipping truck to an appropriate receiving or shipping dock. In the procedure of assigning receiving dock, there are two possible situations based on docking status of the corresponding shipping truck. Therefore, the procedure checks the docking status of the corresponding shipping truck. As a result, if the corresponding shipping truck is not docking in a shipping dock, the procedure finds a pair of available receiving and shipping docks with the minimum travel time (distance) and assigns the receiving truck to the selected receiving dock. In case that the corresponding shipping truck is already in a shipping dock, the procedure finds a receiving dock with minimum distance from the shipping dock which the shipping truck is docking in. Similarly to the receiving dock assignment procedure, shipping dock assignment procedure works in the same way and assigns a shipping dock with minimum distance from the receiving dock which the receiving truck is docking in.

Receiving dock assignment procedure (procedure 6) and shipping dock assignment procedure (procedure 7) are presented in sections 5.2.9 and 5.2.10 respectively.

5.2.2 Truck timing
Based on assumptions, only one unit of product can be unloaded or loaded at any given time. Therefore, it is important to calculate right timing for both receiving and shipping trucks in order to minimize the operation time and also to avoid any product jam in the movement system. As there are multiple numbers of receiving and shipping docks in this model, it is possible that for example a shipping truck will receive needed products from two or more receiving trucks at the same time. As a result, it is necessary to adjust the time that the receiving trucks will send the products to the shipping truck in order to avoid multiple numbers of products arriving at the shipping dock at the same time. However, the scheduling algorithm tries itself to schedule receiving and shipping trucks so that different receiving trucks in receiving docks at the same time will transfer products to different shipping trucks in the shipping docks. This provides a good utilization of multiple receiving and shipping docks.

Timing procedure has been created to adjust the timing for receiving trucks transferring load to the temporary storage and to the shipping trucks as well as the temporary storage transferring products to the shipping trucks.

This procedure compares the time at which the receiving truck can unload products \( (t_{r,un,\text{m}}) \) and the time at which the shipping truck can load products \( (t_{s,\text{m}}) \) as well as the
time at which the temporary storage can receive or send products ($I_{\text{storage}}$). In case of receiving trucks which are not finished with their tasks, the value for $I_{r_{\text{loc}}}$ means the time at which the receiving truck can start unloading products. In other words, this is the time at which the receiving truck finishes its previous unloading of products and is free and can unload another set of products. Similarly, $I_{s_{\text{loc}}}$ has the same meaning for the shipping truck and it is the time at which the shipping truck can load products. Furthermore, this procedure takes into account travel time for products from the receiving truck to the shipping truck and utilizes this time as well.

This procedure is organized in three sub-procedures. Sub-procedure 1 is for the case when product transfer is from a receiving truck to a shipping truck. If product transfer is from a receiving truck to the temporary storage, the sub-procedure 2 is used and if product transfer is from the temporary storage to a shipping truck, then sub-procedure 3 is used for timing. *Timing procedure (Procedure 8)* is presented in section 5.2.11.

### 5.2.3 Scheduling procedure

Steps of the scheduling procedure are:

**Step 1:**
Initialize unscheduled receiving and shipping trucks’ data.

**Step 2:**
Find the first pair of unscheduled receiving and shipping trucks with the highest number of direct product transfer from the receiving to the shipping truck ($\max \sum_{k=1}^{p} T_{ijk}^{\text{temp}}$ for all $i, j$) and transfer the products from the procedure 2.

**Step 3:**
Calculate the number of units of products remaining loaded in the last selected receiving truck ($r_{\text{last}}$) and the number of units of products still needed by the last selected shipping truck ($s_{\text{last}}$).

Number of units of products in the last selected receiving truck = $\sum_{k=1}^{p} p_{r_{\text{last}}k}^{rs}$,

Number of units of products needed by the last selected shipping truck = $\sum_{k=1}^{p} p_{s_{\text{last}}k}^{rs}$.

Based on the results, do the following.

If $\sum_{k=1}^{p} p_{r_{\text{last}}k}^{rs} = 0$ and $\sum_{k=1}^{p} p_{s_{\text{last}}k}^{rs} = 0$ then go to step 4a,

If $\sum_{k=1}^{p} p_{r_{\text{last}}k}^{rs} = 0$ and $\sum_{k=1}^{p} p_{s_{\text{last}}k}^{rs} > 0$ then go to step 4b,
If \( \sum_{k=1}^{P} p^{r}_{lastk} > 0 \) and \( \sum_{k=1}^{P} p^{s}_{lastk} = 0 \) then go to step 4c,

If \( \sum_{k=1}^{P} p^{r}_{lastk} > 0 \) and \( \sum_{k=1}^{P} p^{s}_{lastk} > 0 \) then go to step 4d.

**Step 4:**

**Step 4a:**

The last selected receiving truck (\( r^{last} \)) leaves the receiving dock \( r^{d}_{last} \),

\[
R_{remaining} = R_{remaining} - \{ r^{r}_{last} \}
\]

\[
A^{rd} = A^{rd} + \{ d^{receiving}_{m} \}, \quad \text{where } m = r^{d}_{last} 
\]

\[
l^{rd}_{m} = l^{r}_{r_{last}}, \quad \text{where } m = r^{d}_{last} 
\]

and the last selected shipping truck (\( s^{last} \)) leaves the shipping dock \( s^{d}_{last} \).

\[
S_{remaining} = S_{remaining} - \{ s^{s}_{last} \}
\]

\[
A^{sd} = A^{sd} + \{ d^{shipping}_{n} \}, \quad \text{where } n = s^{d}_{last} 
\]

\[
l^{sd}_{n} = l^{s}_{s_{last}}, \quad \text{where } n = s^{d}_{last} 
\]

Find another pair of receiving and shipping trucks with the highest number of product transfer from *procedure 2*.

Go to step 5.

**Step 4b:**

The last selected receiving truck (\( r^{last} \)) leaves the receiving dock \( r^{d}_{last} \),

\[
R_{remaining} = R_{remaining} - \{ r^{r}_{last} \}
\]

\[
A^{rd} = A^{rd} + \{ d^{receiving}_{m} \}, \quad \text{where } m = r^{d}_{last} 
\]

\[
l^{rd}_{m} = l^{r}_{r_{last}}, \quad \text{where } m = r^{d}_{last} 
\]

Find an unscheduled receiving truck with the highest number of product transfer to the last selected shipping truck (\( s^{last} \)) from *procedure 3*.

Go to step 5.

**Step 4c:**

The last selected shipping truck (\( s^{last} \)) leaves the shipping dock \( s^{d}_{last} \).

\[
S_{remaining} = S_{remaining} - \{ s^{s}_{last} \}
\]

\[
A^{sd} = A^{sd} + \{ d^{shipping}_{n} \}, \quad \text{where } n = s^{d}_{last} 
\]

\[
l^{sd}_{n} = l^{s}_{s_{last}}, \quad \text{where } n = s^{d}_{last} 
\]

Find an unscheduled shipping truck with the highest number of products receiving from the last selected receiving truck (\( r^{last} \)) from *procedure 4*.

Go to step 5.
Step 4d:
Check availability of the receiving and shipping docks.

\[
\begin{align*}
\text{If} & \quad |A^{rd}| > 0 \quad \text{and} \quad |A^{sd}| > 0 & \text{then go to step 4d-1}, \\
\text{If} & \quad |A^{rd}| > 0 \quad \text{and} \quad |A^{sd}| = 0 & \text{then go to step 4d-2}, \\
\text{If} & \quad |A^{rd}| = 0 \quad \text{and} \quad |A^{sd}| > 0 & \text{then go to step 4d-3}, \\
\text{If} & \quad |A^{rd}| = 0 \quad \text{and} \quad |A^{sd}| = 0 & \text{then go to step 4d-4}.
\end{align*}
\]

Step 4d-1:
Find another pair of unscheduled receiving and shipping trucks with the highest number of product transfer from \textit{procedure 2}.
Go to step 5.

Step 4d-2:
Find an unscheduled receiving truck with the highest number of products transferring to the last selected shipping truck ($s_{\text{last}}$) from \textit{procedure 3}.
Go to step 5.

Step 4d-3:
Find an unscheduled shipping truck with the highest number of products receiving from the last selected receiving truck ($r_{\text{last}}$) from \textit{procedure 4}.
Go to step 5.

Step 4d-4:
Check all the docking receiving and shipping trucks to find any possible product transfer from the docking receiving trucks to the docking shipping trucks from \textit{procedure 5}.

Step 5:
Check both remaining receiving trucks and remaining shipping trucks’ sets.

\[
\begin{align*}
\text{If} & \quad |R^{\text{remaining}}| = 0 \quad \text{and} \quad |S^{\text{remaining}}| = 0 & \text{then go to step 7}, \\
\text{If} & \quad |R^{\text{remaining}}| = 0 \quad \text{and} \quad |S^{\text{remaining}}| > 0 & \text{then go to step 6}, \\
\text{If} & \quad |R^{\text{remaining}}| > 0 \quad \text{and} \quad |S^{\text{remaining}}| > 0 & \text{then go to step 3}.
\end{align*}
\]

Step 6:
In this point, all the remaining shipping trucks will receive products needed from the temporary storage from \textit{procedure 9}.
Go to step 5.

Step 7:
Calculate the total operation time and end the scheduling algorithm.
Total operation time is equal to the time at which the last shipping truck leaves the shipping dock minus the time at which the first receiving truck enters the receiving dock.
\[ T = \text{Max} \left( I_{jn}^s \; \text{for all} \; s_j \in S^{\text{scheduled}} \; \text{and where} \; n = s_j^d \right) - \text{Min} \left( d_{im}^r \; \text{for all} \; r_i \in R^{\text{scheduled}} \; \text{and where} \; m = r_i^d \right) \]

The flowchart of the scheduling procedure is shown in figures 5-1 to 5-3.
Step 1: Initialize receiving and shipping trucks data.

Step 2: Find first pair of unscheduled receiving and shipping trucks with highest number of direct product transfer from the receiving to the shipping truck from the procedure 2.

Step 3: Update the list of products loaded in the last scheduled receiving truck and the last scheduled shipping truck to see whether the receiving truck is fully unloaded or not, and similarly whether the shipping truck is fully loaded or not.

D1. Are both the receiving and the shipping trucks satisfied?
   Yes: Step 4a: Both the receiving and the shipping truck leave the receiving and the shipping dock respectively.
   No: Find next pair of receiving and shipping trucks with the highest number of product transfer from procedure 2.

D2. Is the receiving truck satisfied but the shipping truck not satisfied?
   Yes: To Step 5.
   No: Find an unscheduled receiving truck with the highest number of products transfer to the last selected shipping truck from procedure 3.

Figure 5-1. Flowchart - Scheduling algorithm.
D3. Is the receiving truck not satisfied but the shipping truck is satisfied?

No

Yes

Step 4c:
The shipping truck leaves the shipping dock.

Find an unscheduled shipping truck with the highest number of products receiving from the last selected receiving truck from procedure 4.

To Step 5.

Step 4d:
Neither the receiving nor the shipping truck is satisfied. Check for availability of receiving and shipping docks.

D4. Are there both receiving and shipping docks available?

Yes

Step 4d-1:
Find another pair of unscheduled receiving and shipping trucks with the highest number of product transfer from procedure 2.

To Step 5.

No

Step 4d-2:
Find the next unscheduled receiving truck with the highest number of product transfer to the last scheduled shipping truck from procedure 3.

To Step 5.

D5. Is there a receiving dock available but no shipping dock available?

Yes

To D6.

No

To Step 5.

Figure 5-2. Flowchart - Scheduling algorithm (continue).
From D5.

D6. Is there no receiving dock available but a shipping dock available?

Yes

Step 4d-3: Find an unscheduled shipping truck with the highest number of products receiving from the last selected receiving truck from procedure 4.

No

Step 4d-4: There are neither receiving nor shipping docks available. Check all the docking receiving and shipping trucks to find any possible product transfer from the receiving trucks to the shipping trucks from procedure 5.

Step 5: Check both remaining receiving trucks and remaining shipping trucks' sets.

D7. Are both sets empty?

Yes → To Step 7.

No

D8. Is the remaining receiving trucks' set empty but the remaining shipping trucks' set not empty?

Yes → Step 6: Remaining shipping trucks receive needed products from the temporary storage from procedure 9.

No

D9. Are both sets not empty?

No → To Step 3.

Yes → To Step 5.

Step 6: Remaining shipping trucks receive needed products from the temporary storage from procedure 9.

Step 7: Calculate the total operation time.

End of scheduling

Figure 5-3. Flowchart - Scheduling algorithm (continue).
5.2.4 Procedure 1 – Update product transfer

In order to find a pair of receiving and shipping trucks with maximum transfer, it is firstly necessary to calculate all the possible product transfer from each receiving truck to each shipping truck ($\sum_{k=1}^{p} T_{ijk}^{\text{temp}}$ for all $i, j$).

$T_{ijk}^{\text{temp}}$ for each receiving and shipping truck is calculated as:

For all $i, j, k$

$$\begin{cases} 
T_{ijk}^{\text{temp}} = p_{jk}^{rs}, & \text{if } p_{jk}^{rs} \geq p_{ik}^{rr} \\
T_{ijk}^{\text{temp}} = p_{ik}^{rr}, & \text{otherwise}
\end{cases}$$

The flowchart of procedure 1 is shown in figure 5-4.
Procedure 1
Update product transfer

Do the following for each receiving and shipping truck.

Compare the number of products of each type loaded in each receiving truck with the number of products of each type needed by each shipping truck.

Is the number of loaded products in a receiving truck bigger than the number of needed products by a shipping truck?

Yes

Set the number of possible transfer from the receiving truck to the shipping truck equal to the number of products needed by the shipping truck.

No

Set the number of possible transfer from the receiving truck to the shipping truck equal to the number of products loaded in the receiving truck.

End of procedure 1

Figure 5-4. Flowchart - Update product transfer procedure.
5.2.5 Procedure 2 – Finding a pair of unscheduled trucks

This procedure finds a pair of unscheduled receiving and shipping trucks with the highest number of product transfer from the receiving to the shipping truck and schedules them.

Steps of this procedure are:

**Step 1:**
Update list of possible product transfers from receiving trucks to shipping trucks from *procedure 1*.

**Step 2:**
Select a pair of unscheduled receiving and shipping trucks with the highest number of product transfer from the receiving to the shipping truck.

For all \( r_i \in R_{\text{remaining}}, s_j \in S_{\text{remaining}} \), find the maximum value of \( \sum_{k=1}^{P} T_{ijk} \).

**Step 3:**
Set the selected trucks as the last selected receiving and the last selected shipping truck.

Let \( r_{last} = i \) and \( s_{last} = j \).

**Step 4:**
If the selected receiving truck is not already in a receiving dock \( (r_{\text{last}} \not\in R_{\text{docking}}) \), then do the following:

- Add the receiving truck to the list of docking receiving trucks.
  \[ R_{\text{docking}} = R_{\text{docking}} + \{ r_{\text{last}} \} \]
- Add the receiving truck to the end of scheduled receiving trucks’ list.
  \[ R_{\text{scheduled}} = \{ r_{\text{last}} \} \]
- Assign a receiving dock to the receiving truck using the receiving dock assignment procedure (*procedure 6*).

**Step 5:**
If the selected shipping truck is not already in a shipping dock \( (s_{\text{last}} \not\in S_{\text{docking}}) \), then do the following:

- Add the shipping truck to the list of docking shipping trucks.
  \[ S_{\text{docking}} = S_{\text{docking}} + \{ s_{\text{last}} \} \]
- Add the shipping truck to the end of scheduled shipping trucks’ list.
  \[ S_{\text{scheduled}} = \{ s_{\text{last}} \} \]
- Assign the shipping truck to a shipping dock using the shipping dock assignment procedure (*procedure 7*).
Step 6:
Transfer the products from the last selected receiving truck to the last selected shipping truck and update the number of remaining products in the receiving truck and the number of products needed by the shipping truck.

\[
\sum_{k=1}^{p} (T_{r_{\text{last}}_{k}} = T_{\text{temp}}_{r_{\text{last}}_{k}}),
\]

\[
\sum_{k=1}^{p} (p'_{r_{\text{last}}_{k}} = p'_{r_{\text{last}}_{k}} - T_{r_{\text{last}}_{k}}),
\]

\[
\sum_{k=1}^{p} (p'_{s_{\text{last}}_{k}} = p'_{s_{\text{last}}_{k}} - T_{s_{\text{last}}_{k}})
\]

Step 7:
Update timing for both the last selected receiving and the last selected shipping truck using the timing procedure (procedure 8).

The flowchart of procedure 2 is shown in figure 5-5.
Procedure 2

Step 1: Update list of possible transfers between receiving and shipping trucks from "procedure 1."

Step 2: Find the receiving truck and the shipping truck which have the highest number of product transfer from the receiving to the shipping truck.

Step 3: Choose the receiving truck as the last selected receiving truck and the shipping truck as the last selected shipping truck.

D1. Is the last selected receiving truck already in a receiving dock?

   No -> Step 4: Add the last selected receiving truck to the receiving truck docking sequence.

   Yes -> Assign the receiving truck to an available receiving dock from "procedure 6."

D2. Is the last selected shipping truck already in a shipping dock?

   No -> Step 5: Add the last selected shipping truck to the shipping truck docking sequence.

   Yes -> Assign the shipping truck to an available shipping dock from "procedure 7."

Step 6: Transfer the products from the last selected receiving truck to the last selected shipping truck.

Step 7: Update timing for both the receiving and the shipping trucks from "procedure 8."

End of procedure 2

Figure 5-5. Flowchart - Finding a pair of unscheduled trucks procedure.
5.2.6 Procedure 3 - Finding a receiving truck to schedule

This procedure looks for an unscheduled receiving truck with the highest number of product transfer to the last selected shipping truck. Steps for this procedure are:

**Step 1:**
Update list of possible product transfers from receiving trucks to shipping trucks from *procedure 1*.

**Step 2:**
Check the temporary storage for products needed by the last selected shipping truck. If there is any product that can be transferred from the temporary storage to the shipping truck ($\sum_{k=1}^{P} T_{r_{last}}^{fromST} > 0$) go to step 3 otherwise, go to step 5.

**Step 3:**
Transfer the products from the temporary storage to the shipping truck using *procedure 9*.

**Step 4:**
Update timing for the last selected shipping truck and for the temporary storage using the timing procedure (*procedure 8*) and end the procedure.

**Step 5:**
Find an unscheduled receiving truck with the highest number of product transfer to the last selected shipping truck.

For all $r_i \in R^{remaining}$, find the maximum value of $\sum_{k=1}^{P} T_{r_{last}}^{temp} k$.

**Step 6:**
Set selected receiving truck as the last selected receiving truck.

Let $r_{last} = i$.

**Step 7:**
If the selected receiving truck is not already in a receiving dock ($r_{last} \notin R^{docking}$), then do the following.
- Add the receiving truck to the list of docking receiving trucks.
  \[ R^{docking} = R^{docking} + \{ r_{last} \} \]
- Add the receiving truck to the end of scheduled receiving trucks’ list.
  \[ R^{scheduled} = \{ ..., r_{last} \} \]
- Assign the receiving truck to a receiving dock using the receiving dock assignment procedure (*procedure 6*).

**Step 8:**
Transfer the products from the last selected receiving truck to the last selected shipping truck and update the number of remaining products in the receiving truck and the number of products needed by the shipping truck.
\[
\sum_{k=1}^{\tilde{p}} (T_{last, last_k} = T_{last, last_k}^{temp}),
\]
\[
\sum_{k=1}^{\tilde{p}} (p'_{r, last_k} - T_{r, last_k}) = p_{r, last_k},
\]
\[
\sum_{k=1}^{\tilde{p}} (p'_{s, last_k} - T_{s, last_k}) = p_{s, last_k}
\]

Step 9: Update the timing for both the last selected receiving and the last selected shipping truck using the timing procedure (procedure 8).

The flowchart of procedure 3 is shown in figure 5-6.
Procedure 3

Step 1: Update possible transfers between receiving and shipping trucks from procedure 1.

Step 2: Check the temporary storage for any product transfer from the temporary storage to the last selected shipping truck.

D1. Is there any product to be transferred from the temporary storage?

Yes:

Step 3: Transfer the products from the temporary storage to the shipping truck from procedure 9.

No:

Step 5: Find the highest number of product transfer from unscheduled receiving trucks to the last selected shipping truck.

Step 6: Choose the receiving truck as the last selected receiving truck and keep the shipping truck as the last selected shipping truck.

D2. Is the last selected receiving truck already in a receiving dock?

No:

Step 7: Add the last selected receiving truck to the receiving truck docking sequence.

Yes:

Assign the receiving truck to an available receiving dock from procedure 6.

Step 8: Transfer the products from the receiving truck to the shipping truck.

Step 9: Update docking times for both the receiving and the shipping trucks from the procedure 9.

End of procedure 3

End of procedure 3

Figure 5-6. Flowchart - Finding a receiving truck to schedule procedure
5.2.7 Procedure 4 - Finding a shipping truck to schedule
This procedure looks for an unscheduled shipping truck with the highest number of products receiving from the last selected receiving truck. Steps for this procedure are:

**Step 1:**
Update list of possible product transfers from receiving trucks to shipping trucks from *procedure 1*.

**Step 2:**
Find an unscheduled shipping truck with highest number of product transfer from the last selected receiving truck to that shipping truck.

\[
\text{For all } s_j \in S^{\text{remaining}}, \text{ find the maximum value of } \sum_{k=1}^{P} T_{\text{temp},jk}.
\]

**Step 3:**
Set the selected shipping truck as the last selected shipping truck.

Let \( s_{\text{last}} = j \).

**Step 4:**
If the selected shipping truck is not already in a shipping dock (\( s_{\text{last}} \not\in S^{\text{docking}} \)), then do the following.

- Add the shipping truck to the list of docking shipping trucks.
  \[
  S^{\text{docking}} = S^{\text{docking}} + \{ s_{\text{last}} \}
  \]
- Add the shipping truck to the end of scheduled shipping trucks’ list.
  \[
  S^{\text{scheduled}} = \{ ..., s_{\text{last}} \}
  \]
- Assign the shipping truck to a shipping dock using the shipping dock assignment procedure (*procedure 7*).

**Step 5:**
Transfer the products from the last selected receiving truck to the last selected shipping truck and update the number of remaining products in the receiving truck and the number of products needed by the shipping truck.

\[
\sum_{k=1}^{P} (T_{r,\text{last},k} = T_{\text{temp},r,\text{last},k}),
\]

\[
\sum_{k=1}^{P} (p_{r,\text{last},k} = p_{r,\text{last},k} - T_{r,\text{last},k}),
\]

\[
\sum_{k=1}^{P} (p_{s,\text{last},k} = p_{s,\text{last},k} - T_{s,\text{last},k})
\]

**Step 6:**
Update the timing for both the last selected receiving and the last selected shipping truck using the timing procedure (*procedure 8*).

The flowchart of *procedure 4* is shown in figure 5-7.
Procedure 4

Step 1: Update possible transfers between receiving and shipping trucks from procedure 2.

Step 2: Find the highest number of product transfer from the last selected receiving truck to all unscheduled shipping trucks.

Step 3: Choose the shipping truck as the last selected shipping truck and keep the receiving truck as the last selected receiving truck.

D1. Is the last scheduled shipping truck already in a shipping dock?

   - No
     - Step 4: Add the last selected shipping truck to the shipping truck docking sequence.

     - Yes
       - Assign the shipping truck to an available shipping dock from procedure 7.

Step 5: Transfer the products from the receiving truck to the shipping truck.

Step 6: Update docking times for both the receiving and the shipping trucks using procedure 8.

End of procedure 4

Figure 5-7. Flowchart - Finding a shipping truck to schedule procedure
5.2.8 Procedure 5 – Finding product transfer between docking trucks

When there is no receiving or shipping dock available, this procedure finds any product transfer from the docking receiving trucks to the docking shipping trucks. Steps of this procedure are:

**Step 1:**
Update list of possible product transfers from receiving trucks to shipping trucks from *procedure 1*.

**Step 2:**
Select a pair of docking receiving and shipping trucks with the highest number of product transfer from the receiving to the shipping truck.

For all \( r_j \in R^{docking}, s_j \in S^{docking} \), find the maximum value of \( \sum_{k=1}^{p} T_{ijk} \).

If there is no transfer from a docking receiving truck to a docking shipping truck, then go to step 6.

**Step 3:**
Set the selected trucks as the last selected receiving and the last selected shipping trucks.

Let \( r_{last} = i \) and \( s_{last} = j \).

**Step 4:**
Transfer the products from the last selected receiving truck to the last selected shipping truck and update the number of remaining products in the receiving truck and the number of products needed by the shipping truck.

\[
\sum_{k=1}^{p} (T_{jlast,k} = T_{temp}) ,
\sum_{k=1}^{p} (p_{r_{last},k}^r = p_{r_{last},k}^r - T_{r_{last},i_{last}}),
\sum_{k=1}^{p} (p_{s_{last},k}^s = p_{s_{last},k}^s - T_{s_{last},s_{last}})
\]

**Step 5:**
Update the timing for both the last selected receiving and the last selected shipping truck using the timing procedure (*procedure 8*).

End of procedure 5.

**Step 6:**
Check the temporary storage for product transfer to the docking shipping trucks. If there are any products to be transferred to the shipping trucks (\( \sum_{k=1}^{p} T_{j_{fromST}} > 0 \), for all \( s_j \in S^{docking} \)), then go to step 11.


Step 7:
Select a receiving truck with minimum number of products remaining loaded and select that receiving truck as leaving receiving truck.

For all $r_i \in R^{docking}$, find the minimum value of $\sum_{k=1}^{p} p_{ik}^{tr}$

Step 8:
Receiving truck $r_i$ transfers all its remaining products to the temporary storage.

$$\sum_{k=1}^{p} (p_{ik}^{ST} = p_{ik}^{tr})$$

$$\sum_{k=1}^{p} (p_{ik}^{tr} = 0)$$

Step 9:
Update the timing for leaving receiving truck and for the temporary storage using the timing procedure (procedure 8).
The receiving truck leaves the receiving dock.

$$R^{docking} = R^{docking} - \{ r_i \}$$

$$R^{remaining} = R^{remaining} - \{ r_i \}$$

$$A^{rd} = A^{rd} + \{ d^{receiving}_m \}, \quad \text{where} \quad m = r_i^{rd}$$

Step 10:
Based on the previous step, there is a receiving dock available now. Therefore, find an unscheduled receiving truck to satisfy the last selected shipping truck from the procedure 3 and end the procedure.

Step 11:
Transfer the products from the temporary storage to the docking shipping trucks using procedure 9.

Step 12:
Update timing for the temporary storage and for the shipping trucks using procedure 8.

The flowchart of procedure 5 is shown in figure 5-8.
Procedure 5

Step 1: Update possible transfers between receiving and shipping trucks from procedure 1.

For all docking receiving and shipping trucks in receiving and shipping docks do the following.

Step 2: Check docking receiving and shipping trucks for any possible product transfer from a receiving truck to a shipping truck.

D1. Is there any pair of trucks with any product transfer?

Yes

Step 3: Find the receiving truck and the shipping truck which have the highest number of product transfer from the receiving truck to the shipping truck.

Step 4: Transfer the products from the receiving truck to the shipping truck.

D2. Is there any product to be transferred from the temporary storage?

Yes

Step 5: Update docking times for both the receiving and the shipping trucks using procedure 8.

No

Step 6: Check the temporary storage for any product transfer from the temporary storage to the docking shipping trucks.

Step 7: Find the docking receiving truck with the minimum number of products remaining in the truck.

Step 8: Transfer the products from the receiving truck to the temporary storage.

Step 9: Update docking times for the receiving truck using procedure 8 and move the receiving truck out of the receiving dock.

Step 10: Since there is a receiving dock available, find another unscheduled receiving truck to satisfy the last selected shipping truck from procedure 3.

Step 11: Transfer the products from the temporary storage to the docking shipping trucks from procedure 3.

Step 12: Update timing for the shipping trucks and for the temporary storage from procedure 8.

End of procedure 5

Figure 5-8. Flowchart - Finding product transfer between docking trucks procedure
5.2.9 Procedure 6 – Receiving dock assignment

Step 1:
Check if the associated shipping truck is already in a shipping dock or no. If the shipping truck is not in a shipping dock ($s_{s_{low}} \notin S_{docking}$) then go to step 2, otherwise go to step 3.

Step 2:
Find a pair of available receiving and shipping docks with the minimum travel time between them.

For all $d_{m}^{receiving} \in A^{rd}, d_{n}^{shipping} \in A^{sd}$ find minimum value for $W_{mn}$

Go to step 4.

Step 3:
Find a receiving dock with minimum distance from the shipping dock which the associated shipping truck is docking in.

For all $d_{m}^{receiving} \in A^{rd}, n = s_{s_{last}}$ find minimum value for $W_{mn}$

Step 4:
Assign the last selected receiving truck ($r_{last}$) to the receiving dock $m$ selected from the previous step and update available receiving docks and docking receiving trucks’ set.

\[
\begin{align*}
    r_{\text{last}}^{d} &= m, \\
    R_{\text{docking}} &= R_{\text{docking}} + \{r_{\text{last}}^{d}\}, \\
    A^{rd} &= A^{rd} - \{d_{m}^{receiving}\}, \quad \text{where } m = r_{\text{last}}^{d}
\end{align*}
\]

Step 5:
Set the docking time for the last selected receiving truck entering the selected receiving dock. Docking time for the receiving truck is the time in which the receiving dock $m$ became available plus the truck change time.

\[
\begin{align*}
    d_{r_{\text{last}}^{d}}^{r} &= l_{m}^{rd} + D, \\
    l_{r_{\text{last}}^{d}}^{r} &= d_{r_{\text{last}}^{d}}^{r}
\end{align*}
\]

The flowchart of procedure 6 is shown in figure 5-9.

When the last selected receiving truck is fully unloaded and leaves the receiving dock, following updates are made by the algorithm

\[
\begin{align*}
    A^{rd} &= A^{rd} + \{d_{m}^{receiving}\}, \quad \text{where } m = r_{\text{last}}^{d} \\
    l_{m}^{r} &= l_{r_{\text{last}}^{d}}^{r}, \quad \text{where } m = r_{\text{last}}^{d}
\end{align*}
\]
Procedure 6

Step 1: Check docking status of the associated shipping truck.

D1. Is the associated shipping truck already in a shipping dock?

No

Step 2: For all available receiving and shipping docks find a pair of receiving and shipping docks with the minimum distance.

Yes

Step 3: From all available receiving docks, find the receiving dock with the minimum distance from the shipping dock which the associated shipping truck is docking in.

Step 4: Assign the receiving truck to the selected receiving dock.

Step 5: Set the docking time for the last selected receiving truck.

End of procedure 6

Figure 5-9. Flowchart - Receiving dock assignment procedure.
5.2.10 Procedure 7 – Shipping dock assignment

Step 1:
Check if the associated receiving truck is already in a receiving dock or no. If the receiving truck is not in a receiving dock \( r_{last} \in R^{\text{docking}} \) then go to step 2, otherwise go to step 3.

Step 2:
Find a pair of available receiving and shipping docks with the minimum travel time between them.

For all \( d_{m}^{\text{receiving}} \in A^{rd}, d_{n}^{\text{shipping}} \in A^{nd} \), find minimum value for \( W_{mn} \)

Go to step 4.

Step 3:
Find a shipping dock with minimum distance from the receiving dock which the associated receiving truck is docking in.

For all \( d_{m}^{\text{receiving}} \in A^{rd}, d_{n}^{\text{shipping}} \in A^{nd} \), find minimum value for \( W_{mn} \)

Step 4:
Assign the last scheduled shipping truck \( s^{\text{last}} \) to the shipping dock \( n \) selected from the previous step and update available shipping docks and docking shipping trucks’ set.

\[
\begin{align*}
s^{\text{last}} &= n, \\
S^{\text{docking}} &= S^{\text{docking}} + \{ s^{\text{last}} \}, \\
A^{\text{nd}} &= A^{\text{nd}} - \{ d_{n}^{\text{shipping}} \}, \quad \text{where } n = s^{\text{last}}_{d}.
\end{align*}
\]

Step 5:
Set the docking time for the last selected shipping truck entering the selected shipping dock. Docking time for the shipping truck is the time in which the shipping dock \( n \) became available plus the truck change time.

\[
\begin{align*}
d_{s^{\text{last}}_{n}} &= l_{n}^{\text{td}} + D \\
l_{s^{\text{last}}_{n}} &= d_{s^{\text{last}}_{n}}
\end{align*}
\]

The flowchart of procedure 7 is shown in figure 5-10

When the last selected shipping truck is fully loaded and leaves the shipping dock, following updates are made by the algorithm

\[
\begin{align*}
A^{\text{nd}} &= A^{\text{nd}} + \{ d_{n}^{\text{shipping}} \}, \quad \text{where } n = s^{\text{last}}_{d},
\end{align*}
\]

\[
\begin{align*}
l_{n}^{\text{td}} &= l_{s^{\text{last}}_{n}}, \quad \text{where } n = s^{\text{last}}_{d}
\end{align*}
\]
Procedure 7

Step 1: Check docking status of the associated receiving truck.

D1. Is the associated receiving truck already in a receiving dock?

Yes

Step 2: For all available receiving and shipping docks find a pair of receiving and shipping docks with the minimum distance.

No

Step 3: From all available shipping docks, find the shipping dock with the minimum distance from the receiving dock which the associated receiving truck is docking in.

Step 4: Assign the shipping truck to the selected shipping dock.

Step 5: Set the docking time for the last selected shipping truck.

End of procedure 7

Figure 5-10. Flowchart - Shipping dock assignment procedure.
5.2.11 Procedure 8 – Truck timing

Sub-procedure 1

This is the case where a receiving truck is transferring products to a shipping truck. Steps for this sub-procedure are:

Step 1:
Check which one of the receiving or shipping trucks will be able to send or receive products later with taking into account the travel time between the receiving dock and the shipping dock.

If \( I_{r_{m}}^r \geq I_{s_{n}}^s - W_{mn} \) where \( m = r_{d_{s_{m}}} \), then go to step 2; otherwise go to step 3.

Step 2:
In this step as the receiving truck will be available to send products later than the time at which the shipping truck will be able to receive them, adjust the timing for trucks as:

\[
I_{r_{m}}^r = I_{r_{m}}^r + \sum_{k=1}^{p} T_{r_{d_{s_{m}}} s_{r_{m}}} h_k , \text{ where } m = r_{d_{s_{m}}}, n = s_{d_{s_{m}}}
\]

In equation above, \( \sum_{k=1}^{p} T_{r_{d_{s_{m}}} s_{r_{m}}} h_k \) calculates the total time which it takes to unload or load the products that transfer from the receiving truck to the shipping truck taking into account different loading and unloading time for each product type.

\[
I_{s_{n}}^s = I_{r_{m}}^r + W_{mn} , \text{ where } m = r_{d_{s_{m}}}, n = s_{d_{s_{m}}}
\]

Step 3:
As in this case the shipping truck will be ready later than the receiving truck, this step will adjust the timing based on the shipping truck’s availability as following.

\[
I_{r_{m}}^r = I_{r_{m}}^r - W_{mn} + \sum_{k=1}^{p} T_{r_{d_{s_{m}}} s_{r_{m}}} h_k , \text{ where } m = r_{d_{s_{m}}}, n = s_{d_{s_{m}}}
\]

\[
I_{s_{n}}^s = I_{r_{m}}^r + W_{mn} , \text{ where } m = r_{d_{s_{m}}}, n = s_{d_{s_{m}}}
\]


**Sub-procedure 2**

This sub-procedure is used in the case that a receiving truck is transferring products to the temporary storage. Steps for this sub-procedure are:

**Step 1:**
Check which one of the receiving trucks or the temporary storage will be able to send or receive products later taking into account the travel time between the receiving truck and the temporary storage.

\[
I_{r_{d_{m}}}^{r} \geq I_{m}^{storage} - W_{m}^{rs} \quad \text{where } m = r_{d_{m}} \text{ then go to step 2, otherwise go to step 3.}
\]

**Step 2:**
In this step, as the receiving truck will be available to send products later than the time at which the temporary storage will be able to receive them, adjust the timing for the receiving truck and the temporary storage as:

\[
I_{r_{d_{m}}}^{r} = I_{r_{d_{m}}}^{r} + \sum_{k=1}^{p} T_{r_{d_{m}}k}^{rST} h_{k}, \quad \text{where } m = r_{d_{m}}
\]

\[
I_{m}^{storage} = I_{m}^{r} + W_{m}^{rs}, \quad \text{where } m = r_{d_{m}}
\]

**Step 3:**
As in this case the temporary storage will be ready later than the receiving truck, this step will adjust the timing based on the temporary storage’s availability as following.

\[
I_{r_{d_{m}}}^{r} = I_{m}^{storage} - W_{m}^{rs} + \sum_{k=1}^{p} T_{r_{d_{m}}k}^{rST} h_{k}, \quad \text{where } m = r_{d_{m}}
\]

\[
I_{m}^{storage} = I_{r_{d_{m}}}^{r} + W_{m}^{rs}, \quad \text{where } m = r_{d_{m}}
\]

**Sub-procedure 3**

This is the case where there is a product transfer from the temporary storage to a shipping truck. Steps for this sub-procedure are:

**Step 1:**
Similarly to the previous sub-procedures, this sub-procedure checks which one of the temporary storages or the shipping truck will be able to send or receive products later taking into account the travel time between the temporary storage and the shipping truck.

\[
I_{m}^{storage} \geq I_{s_{d_{n}}}^{s} - W_{n}^{sn} \quad \text{where } n = s_{d_{n}} \text{ then go to step 2, otherwise go to step 3.}
\]

**Step 2:**
In this step, as the temporary storage will be available to send products later than the time at which the shipping truck will be able to receive them, adjust the timing for the temporary storage and the shipping truck as:
\[ I_{\text{storage}} = I_{\text{storage}} + \sum_{k=1}^{p} T_{\text{fromST}}^{k} h_{k}, \quad \text{where } n = s_{i_{\text{last}}}^{d} \]
\[ I_{\text{stor}_{n}}^{s} = I_{\text{storage}} + W_{n}^{sn}, \quad \text{where } n = s_{i_{\text{last}}}^{d} \]

**Step 3:**
As in this case, the shipping truck will be ready to receive the products later than the temporary storage will be ready to transfer products, this step will adjust the timing based on the shipping truck’s availability as following.

\[ I_{\text{storage}} = I_{\text{stor}_{n}}^{s} - W_{n}^{sn} + \sum_{k=1}^{p} T_{\text{fromST}}^{k} h_{k}, \quad \text{where } n = s_{i_{\text{last}}}^{d} \]
\[ I_{\text{stor}_{n}}^{s} = I_{\text{storage}} + W_{n}^{sn}, \quad \text{where } n = s_{i_{\text{last}}}^{d} \]

The flowchart of procedure 8 is shown in figure 5-11 and 5-12.
Procedure 8

D1. Is the transfer from a receiving truck to a shipping truck?

- Yes → Sub-procedure 1

D2. Will the receiving truck be ready later than the shipping truck?

- Yes → Step 2: Calculate departure or availability time for both receiving and shipping truck based on the receiving truck’s availability
- No → D3. Is the transfer from a receiving truck to the temporary storage?

- Yes → End of procedure 8
- No → Sub-procedure 2

D3. Is the transfer from a receiving truck to the temporary storage?

- Yes → D4. Will the receiving truck be ready later than the temporary storage?

- Yes → Step 2: Calculate departure or availability time for both receiving and temporary storage based on the receiving truck’s availability
- No → End of procedure 8
- No → Step 3: Calculate departure or availability time for both receiving and temporary storage based on the temporary storage’s availability

To D5.

Figure 5-11. Flowchart – Timing procedure.
D5. Will the temporary storage be ready later than the shipping truck?

Sub-procedure 3

Step 2: Calculate departure or availability time for both the temporary storage and the shipping truck based on the temporary storage’s availability

Step 3: Calculate departure or availability time for both the temporary storage and the shipping truck based on the shipping truck’s availability

End of procedure 8

Figure 5-12. Flowchart – Timing procedure (continue).
5.2.12 Procedure 9 – Product transfer from temporary storage

This procedure transfers products needed by shipping trucks from the temporary storage. Steps for this procedure are:

Step 1:
If the shipping truck is not already in a shipping dock ($s_{last} \not\in S^{docking}$), then do the following:

- Add the shipping truck to the list of docking shipping trucks.
  \[ S^{docking} = S^{docking} + \{ s_{last} \} \]
- Add the shipping truck to the end of scheduled shipping trucks’ list.
  \[ S^{scheduled} = \{ \ldots, s_{last} \} \]
- Assign a shipping dock to the shipping truck using the shipping dock assignment procedure (procedure 7).

Step 2:
Transfer all the products needed by the shipping truck from the temporary storage.

\[ T_{fromST}^{s_{last}} = p_{s_{last}}^{ts} \quad \text{for all } k=1, \ldots, P \]

\[ p_{k}^{ts} = p_{k}^{ts} - p_{s_{last}}^{ts} \quad \text{for all } k=1, \ldots, P \]

Step 3:
Update timing for the shipping truck and for the temporary storage using the timing procedure (procedure 8)

Step 4:
The shipping truck leaves the shipping dock.

\[ S^{docking} = S^{docking} - \{ s_{last} \} \]
\[ S^{remaining} = S^{remaining} - \{ s_{last} \} \]
\[ A^{sd} = A^{sd} + \{ d_{n}^{shipping} \}, \quad \text{where } n = s_{s_{last}}^{d} \]

The flowchart of procedure 9 is shown in figure 5-13
Procedure 9

D1. Is the last selected shipping truck already in a shipping dock?

No

Step 1:
Add the last selected shipping truck to the shipping truck docking sequence.

Assign the shipping truck to an available shipping dock from procedure 7.

Yes

Step 2:
Transfer products needed by the shipping truck from the temporary storage to the shipping truck.

Step 3:
Update docking times for both the receiving and shipping trucks.

Step 4:
The shipping truck leaves the shipping dock.

End of procedure 9

Figure 5-13. Flowchart - Product transfer from temporary storage procedure.
6 Results and conclusions

In this chapter, results of using the mathematical model and the heuristic algorithm on test data sets presented in appendix B are discussed. This chapter is organized in two main parts. First part is presentation of result for test data sets used for both mathematical model and heuristic algorithm and the comparison of the results. This part will help to evaluate the performance of the heuristic algorithm. Second part will however present only the results for heuristic algorithm; in this part, test data sets are in larger scale as the generated heuristic algorithm has almost no limitations.

6.1 Results of mathematical model

In this section, results of mathematical model for test data sets 1 to 5 from appendix B as well as the comparison of these results with the results of heuristic algorithm for same the data sets are presented in tables 6-1 to 6-3 for different values of truck changing times. In addition, table 6-5 presents the result of mathematical model and heuristic algorithm for data set 4 with different number of receiving and shipping docks.

In this chapter truck changing times of 50, 100, and 150 were used in both mathematical model and heuristic algorithm for data sets 1 to 5 and truck changing time of 50 was used for the data sets 6 to 10.
Table 6-1. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 50.

<table>
<thead>
<tr>
<th>Set no.</th>
<th>Results for the heuristic algorithm</th>
<th>Docking sequence</th>
<th>Makespan</th>
<th>Computation time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R: 3, 2, 1</td>
<td>S: 1, 2</td>
<td>970</td>
<td>0.047</td>
</tr>
<tr>
<td>2</td>
<td>R: 3, 1, 2</td>
<td>S: 1, 2</td>
<td>1472</td>
<td>0.084</td>
</tr>
<tr>
<td>3</td>
<td>R: 3, 1, 2</td>
<td>S: 1, 2</td>
<td>2262</td>
<td>0.218</td>
</tr>
<tr>
<td>4</td>
<td>R: 1, 2, 3, 4</td>
<td>S: 1, 1, 1, 1</td>
<td>2140</td>
<td>0.11</td>
</tr>
<tr>
<td>5</td>
<td>R: 1, 3, 2</td>
<td>S: 1, 2, 2, 2</td>
<td>2062</td>
<td>1.062</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set no.</th>
<th>Results for the mathematical model</th>
<th>Docking sequence</th>
<th>Makespan</th>
<th>Total number of products</th>
<th>Number of product types</th>
<th>Number of shipping docks</th>
<th>Number of receiving docks</th>
<th>Number of shipping trucks</th>
<th>Number of receiving trucks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R: 2, 3, 1</td>
<td>S: 1, 2</td>
<td>940</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>R: 2, 3, 1</td>
<td>S: 1, 2</td>
<td>1306</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>R: 3, 1, 2</td>
<td>S: 1, 2</td>
<td>2262</td>
<td>8</td>
<td>8</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>R: 1, 4, 2, 3</td>
<td>S: 1, 2, 2, 2</td>
<td>2054</td>
<td>6</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>R: 1, 3, 2</td>
<td>S: 1, 2, 2</td>
<td>2038</td>
<td>9</td>
<td>9</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Table 6-2. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 100.

<table>
<thead>
<tr>
<th>Set no.</th>
<th>Results for the mathematical model</th>
<th>Results for the heuristic algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Docking sequence</td>
<td>Makespan</td>
</tr>
<tr>
<td>1</td>
<td>R: 2, 3, 1, S: 2, 1, 2</td>
<td>1040</td>
</tr>
<tr>
<td>2</td>
<td>R: 2, 3, 1, S: 2, 1, 2</td>
<td>1406</td>
</tr>
<tr>
<td>3</td>
<td>R: 2, 3, 1, S: 2, 1, 2</td>
<td>2168</td>
</tr>
<tr>
<td>4</td>
<td>R: 2, 3, 1, S: 2, 1, 2</td>
<td>2202</td>
</tr>
<tr>
<td>5</td>
<td>R: 2, 3, 1, S: 2, 1, 2</td>
<td>2188</td>
</tr>
</tbody>
</table>

| Total number of products | 450 | 681 | 828 | 610 | 966 |
| Number of product types | 3 | 3 | 3 | 2 | 4 |
| Number of shipping docks | 2 | 2 | 2 | 1 | 2 |
| Number of receiving docks | 2 | 1 | 1 | 2 | 2 |
| Number of shipping trucks | 3 | 3 | 4 | 3 | 4 |
| Number of receiving trucks | 3 | 3 | 4 | 3 | 5 |

59
Table 6-3. Results of mathematical model and heuristic algorithm for data sets 1 to 5 with truck changing time of 150.

<table>
<thead>
<tr>
<th>Docking sequence</th>
<th>Makespan</th>
<th>构想 sequence</th>
<th>Makespan</th>
<th>构想 sequence</th>
<th>Makespan</th>
<th>构想 sequence</th>
<th>Makespan</th>
<th>构想 sequence</th>
<th>Makespan</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R: 3, 2, 1</td>
<td>1140</td>
<td>R: 3, 1, 2</td>
<td>1572</td>
<td>R: 3, 1, 2</td>
<td>2562</td>
<td>R: 1, 2, 3, 4</td>
<td>2640</td>
<td>R: 1, 3, 2</td>
<td>2246</td>
</tr>
<tr>
<td>S: 1, 2</td>
<td></td>
<td>S: 1, 2</td>
<td></td>
<td>S: 1, 3, 4</td>
<td></td>
<td>S: 3, 3, 1</td>
<td></td>
<td>S: 3, 1, 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.078</td>
<td>0.078</td>
<td>0.11</td>
<td>0.172</td>
<td>0.921</td>
<td>0.921</td>
<td></td>
<td>0.921</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total number of products</th>
<th>450</th>
<th>681</th>
<th>828</th>
<th>610</th>
<th>966</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of product types</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Number of shipping docks</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Number of receiving docks</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Number of shipping trucks</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Number of receiving trucks</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Set no.</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>
It can be seen that the results from the mathematical model and the heuristic algorithm are very close with slight differences in receiving and shipping truck docking sequence. There are also differences in product routing between these two methods. As an example, product routings for data sets 1 and 3 resulted from two methods are presented in table 6-4.

Table 6-4. Product routing results for data sets 1 and 3.

<table>
<thead>
<tr>
<th></th>
<th>Mathematical model</th>
<th>Heuristic algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>To</td>
<td>Quantity</td>
</tr>
<tr>
<td>receiving</td>
<td>shipping</td>
<td></td>
</tr>
<tr>
<td>truck</td>
<td>truck</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>130</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>70</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>140</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>30</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>161</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>53</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>234</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>76</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>203</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6-5, presents the results of comparison between the mathematical model and the heuristic algorithm applied on data set 4 with different number of receiving and shipping docks in order to see the performance of heuristic algorithm with different receiving and shipping docks number.
Table 6-5. Results for data set 4 with different number of docks.

<table>
<thead>
<tr>
<th>Docking sequence</th>
<th>Makespan</th>
<th>Computation time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Heuristic</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R: 1, 2, 3, 4 S: 2, 1, 3</td>
<td>2140</td>
<td>0.11</td>
</tr>
<tr>
<td>R: 1, 3, 2, 4 S: 2, 1, 3</td>
<td>1988</td>
<td>1.453</td>
</tr>
<tr>
<td>R: 1, 2, 3, 4 S: 2, 1, 3</td>
<td>1984</td>
<td>0.109</td>
</tr>
<tr>
<td>R: 1, 3, 2, 4 S: 2, 1, 3</td>
<td>1280</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Docking sequence</th>
<th>Makespan</th>
<th>Number of shipping docks</th>
<th>Number of receiving docks</th>
<th>Set no.</th>
</tr>
</thead>
<tbody>
<tr>
<td>R: 1, 4, 2, 3 S: 3, 2, 1</td>
<td>2054</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>R: 3, 4, 1, 2 S: 1, 3, 2</td>
<td>1934</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>R: 4, 3, 1, 2 S: 1, 3, 2</td>
<td>1880</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>R: 4, 3, 1, 2 S: 1, 3, 2</td>
<td>1256</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
6.2 Results of heuristic algorithm

In this section, results of heuristic algorithm applied on data sets 6 to 10 from appendix B are presented. Data sets used in this section are in larger scale and beyond the limitations of student version of AMPL. Furthermore, the computational time for mathematical model increases dramatically when the number of receiving and shipping trucks increases as well as the number receiving and shipping docks. Therefore, only the heuristic algorithm was used for larger data sets in this research. The results of heuristic algorithm for these data sets are presented in table 6-6. In these data sets, distances between receiving and shipping docks used are presented in table B-6, in addition to the loading and unloading time for products which are presented in table B-7. Number of iterations of the algorithm for each data set presented in the table corresponds to the number of times that the scheduling procedure (see section 5.2) was used in the heuristic algorithm to solve the problem. In addition, the heuristic algorithm also calculates number of products transferring from receiving trucks to the temporary storage as well as the number of products transferring from the temporary storage to shipping truck in contrary to the mathematical model. Total number of units of products passing through the temporary storage calculated by heuristic algorithm is also presented in the table. It can be seen from the results that the heuristic algorithm transfers as few products as possible to the temporary storage.

In cases that receiving and shipping trucks are divided into groups or clusters in the cross docking operation, this algorithm can be used to calculate optimal number of receiving and shipping docks based on preferences of total operation time or total number of products passing through the temporary storage.
<table>
<thead>
<tr>
<th>Set</th>
<th>Number of receiving docks</th>
<th>Number of shipping docks</th>
<th>Number of products</th>
<th>Makespan</th>
<th>Products passing storage</th>
<th>Docking sequence</th>
<th>Number of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>4</td>
<td>3</td>
<td>4447</td>
<td>5313</td>
<td>119</td>
<td>R: 2, 3, 7, 5, 4, 1, 6, 8</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S: 2, 5, 6, 4, 1, 3</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>5</td>
<td>5186</td>
<td>4886</td>
<td>0</td>
<td>R: 2, 8, 5, 1, 7, 6, 3, 4, 9, 10</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S: 9, 3, 4, 1, 2, 6, 5, 8, 7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>6</td>
<td>6</td>
<td>8250</td>
<td>8595</td>
<td>24</td>
<td>R: 12, 11, 3, 7, 2, 1, 4, 9, 5, 10, 8, 6</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S: 1, 2, 3, 4, 10, 6, 5, 8, 9, 7</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>9</td>
<td>5316</td>
<td>3488</td>
<td>0</td>
<td>R: 3, 12, 8, 6, 15, 1, 7, 2, 14, 4, 5, 11, 13, 10, 9</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S: 1, 6, 10, 3, 13, 4, 9, 15, 8, 5, 11, 14, 2, 7, 12</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>5</td>
<td>2530</td>
<td>4162</td>
<td>14</td>
<td>R: 3, 1, 2, 6, 9, 7, 4, 11, 5, 12, 10, 8</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S: 11, 12, 5, 1, 2, 4, 8, 3, 6, 7, 10, 9</td>
<td></td>
</tr>
</tbody>
</table>
6.3 Dock assignment results

In this section, results of the mathematical model and the heuristic algorithm in case of dock door assignment are presented.

Table 6-7 presents dock assignment for data sets 1 to 10 for both receiving and shipping trucks resulted by heuristic algorithm.

Although there is some research conducted regarding dock door assignment problem, there is no standard measurement for the performance of the method used for dock door assignment. Therefore, as a method to measure the performance of the dock assignment solution developed in this research, close shipping docks to each receiving dock is determined and the percentage of products transferred from a receiving dock to its close shipping docks is calculated. Determining close shipping docks to each receiving dock is based on the distance of receiving docks from the shipping trucks. Moreover, as in this research distances between receiving and shipping docks presented in table B-6 in appendix B are applied to all data sets, example of determining close shipping docks illustrated below is valid for all receiving and shipping docks.

![Sample layout of receiving and shipping docks.](image)

According to figure 6-1, shipping docks 1 and 2 are determined as close shipping docks to receiving dock 1. In addition, shipping docks 2, 3, and 4 are close to receiving dock 3 and in case of receiving dock 5, shipping dock 4 is the close dock.

For data sets 6 to 10 which have more than two receiving and shipping docks, percentage of unloaded products into each receiving dock transferred to close shipping docks are
calculated and the results are presented in table 6-8 and table 6-9. As mentioned earlier, this is not a standard performance indicator for dock door assignment but it is effective to evaluate performance of the algorithm.
Table 6-7. Receiving and shipping dock door assignment results for all data sets.

<table>
<thead>
<tr>
<th>Receiving truck</th>
<th>Set</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shipping truck</th>
<th>Set</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Receiving dock</th>
<th>Set</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Set</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>
Table 6-8. Percentage of products transferred to close shipping docks from receiving docks.

<table>
<thead>
<tr>
<th>Data set 6</th>
<th>Percentage of products transferred to close shipping docks from the receiving dock (%)</th>
<th>Percentage of total products unloaded into the receiving dock (%)</th>
<th>Shipping docks considered as close to the receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100.00</td>
<td>65.84</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td></td>
<td>90.75</td>
<td>82.53</td>
<td>3, 4, 5</td>
</tr>
<tr>
<td></td>
<td>84.78</td>
<td>84.78</td>
<td>3, 4, 5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data set 7</th>
<th>Percentage of products transferred to close shipping docks from the receiving dock (%)</th>
<th>Percentage of total products unloaded into the receiving dock (%)</th>
<th>Shipping docks considered as close to the receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100.00</td>
<td>69.49</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td></td>
<td>90.75</td>
<td>82.53</td>
<td>3, 4, 5</td>
</tr>
<tr>
<td></td>
<td>89.02</td>
<td>89.02</td>
<td>3, 4, 5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data set 8</th>
<th>Percentage of products transferred to close shipping docks from the receiving dock (%)</th>
<th>Percentage of total products unloaded into the receiving dock (%)</th>
<th>Shipping docks considered as close to the receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100.00</td>
<td>66.13</td>
<td>1, 2</td>
</tr>
<tr>
<td></td>
<td>90.75</td>
<td>82.53</td>
<td>3, 4, 5</td>
</tr>
<tr>
<td></td>
<td>89.02</td>
<td>89.02</td>
<td>3, 4, 5</td>
</tr>
</tbody>
</table>

Receiving dock

Average

68
Table 6-9. Percentage of products transferred to close shipping docks from receiving docks
(continue)

<table>
<thead>
<tr>
<th>Data set 10</th>
<th>Average of products transferred to close shipping docks from the receiving dock (%)</th>
<th>Percentage of products transferred to close shipping docks from the receiving dock (%)</th>
<th>Percentage of total products unloaded into the receiving dock (%)</th>
<th>Shipping docks considered as close to the receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>72.51 78.67 83.34 83.29 85.79</td>
<td>64.65 63.24 86.52 83.34 83.29</td>
<td>85.79 85.79 82.00 82.00 88.79</td>
<td>20.72 73.30 - 20.72 73.30</td>
</tr>
<tr>
<td>Data set 9</td>
<td>65.92 98.21 92.33 93.58 92.01 99.09 82.00 88.79 20.72</td>
<td>50.43 10.55 14.86 - - - - - -</td>
<td>77.47 - - - - - - - - - - - -</td>
<td>- - - - - - - - - - - -</td>
</tr>
</tbody>
</table>

According to the results, in average over 73% of products unloaded to receiving docks in all data sets are transferred to close shipping docks.

6.4 Conclusions

In this thesis, the focus was on optimization of inbound and outbound truck scheduling with the goal of minimizing total operation time of cross docking. A model of cross docking was developed and two different methods were applied on it in order to find an optimal docking sequence for receiving and shipping trucks and their assignment to receiving and shipping docks, and product routing from receiving to shipping trucks.
The two methods used were mathematical modeling and heuristic algorithm. For the first method, a mixed integer programming model was developed to minimize total operation time. This model was formulated in AMPL modeling language and used to solve small size problems because of the limitations of student version of AMPL and the computational time required by this method. For the second method, a heuristic algorithm was developed to find near optimal solutions fast and was used for problems of larger size. In order to examine the performance of heuristic algorithm, small problems were solved by both mathematical model and the heuristic algorithm. When developing the heuristic algorithm for the scheduling problem, the main strategy to select receiving and shipping trucks to be scheduled was the number of products that can be transferred from a receiving truck to a shipping truck directly. The heuristic algorithm for the scheduling problem minimized the number of product transfer from receiving trucks to the temporary storage by using all available receiving and shipping docks.

Comparison of the results of both methods was also provided; the results from the mathematical model and the heuristic algorithm are very close with slight differences in receiving and shipping truck docking sequence. There are also differences in product routing between these two methods. In addition, the heuristic algorithm also calculated number of products transferring from receiving trucks to the temporary storage as well as the number of products transferring from the temporary storage to shipping truck in contrary to the mathematical model. Total number of units of products passing through the temporary storage calculated by heuristic algorithm is presented and it can be seen that the heuristic algorithm transfers to the temporary storage as few products as possible. Furthermore, in cases that receiving and shipping trucks are divided into groups or clusters in the cross docking operation, heuristic algorithm can be used to calculate optimal number of receiving and shipping docks based on preferences of total operation time or total number of products passing through the temporary storage.

Another issue which was focused on within the thesis was the problem of dock door assignment. As there exists no standard measurement for the performance of the methods used for dock door assignment, close shipping docks to each receiving dock were determined and the percentage of products transferred from a receiving dock to its close shipping docks was calculated as a method to measure the performance of the dock assignment solution. Determining close shipping docks to each receiving dock was based on the distance of receiving docks from the shipping trucks. According to the results, in average over 73% of all products unloaded to receiving docks in all data sets were transferred to close shipping docks. Even though it is not a standard performance indicator for dock door assignment, it proved to be effective to evaluate performance of the algorithm.

As cross docking is getting more popular, it will be necessary to do more research in different areas of cross docking. For truck scheduling process of cross docking operation, there is still a need for more research in order to develop optimal and effective methods to be used for realistic problems.
List of references


Appendix A Mathematical model formulation in AMPL

Model file (Model.txt)

# PARAMETERS
param P; # number of products
param R; # number of receiving tricks
param S; # number of shipping trucks
param M; # number of receiving docks
param N; # number of shipping docks

param loaded{1..R,1..P};
param needed{1..S,1..P};
param delay;
param moving{1..M,1..N};
param BigM;
param loading{1..P};

# INTEGER VARIABLES
var x{1..R,1..S,1..P} integer>=0;

# BINARY VARIABLES
var p{1..R,1..R} binary;
var q{1..S,1..S} binary;
var t{1..R,1..S} binary;
var Ar{1..R,1..M} binary;
var As{1..S,1..N} binary;

# CONTINUOUS VARIABLES
var DR{1..R,1..M}>=0;
var LR{1..R,1..M}>=0;
var DS{1..S,1..N}>=0;
var LS{1..S,1..N}>=0;

# OBJECTIVE FUNCTION
minimize  f : sum {n in 1..N, j in 1..S} LS[j,n] ;

# CONSTRAINTS
subject to c2{i in 1..R}:
    sum {m in 1..M} Ar[i,m] = 1;

subject to c3{m in 1..M}:
    sum {i in 1..R} Ar[i,m] >= 1;

subject to c4{j in 1..S}:
sum {n in 1..N} As[j,n] = 1;

subject to c5{n in 1..N}:
  sum {j in 1..S} As[j,n] >= 1;

subject to c6{i in 1..R, k in 1..P}:
  sum {j in 1..S} x[i,j,k] = loaded[i,k];

subject to c7{j in 1..S, k in 1..P}:
  sum {i in 1..R} x[i,j,k] = needed[j,k];

subject to c8{i in 1..R, j in 1..S, k in 1..P}:
  x[i,j,k] <= BigM * t[i,j];

subject to c9{i in 1..R, m in 1..M}:
  LR[i,m] >= DR[i,m] + Ar[i,m] * sum {k in 1..P} loaded[i,k] * loading[k];

subject to c10{i in 1..R, j in 1..R, m in 1..M : i<>j}:
  DR[j,m] >= LR[i,m] + delay - BigM * (1-p[i,j]);

subject to c11{i in 1..R, j in 1..R, m in 1..M : i<>j}:
  DR[i,m] >= LR[j,m] + delay - BigM * p[i,j];

subject to c12{i in 1..R}:
  p[i,i]=0;

subject to c13{j in 1..S, n in 1..N}:
  LS[j,n] >= DS[j,n] + As[j,n] * sum {k in 1..P} needed[j,k] * loading[k];

subject to c14{i in 1..S, j in 1..S, n in 1..N : i<>j}:
  DS[j,n] >= LS[i,n] + delay - BigM * (1-q[i,j]);

subject to c15{i in 1..S, j in 1..S, n in 1..N : i<>j}:
  DS[i,n] >= LS[j,n] + delay - BigM * q[i,j];

subject to c16{i in 1..S}:
  q[i,i]=0;

subject to c17{i in 1..R, j in 1..S, m in 1..M, n in 1..N}:
  LS[j,n] + BigM * (1-As[j,n]) >= DR[i,m] + moving[m,n] + sum {k in 1..P} x[i,j,k] * loading[k] - BigM * (1-t[i,j]) - BigM * (1-Ar[i,m]);

subject to c18{i in 1..R, j in 1..S, m in 1..M, n in 1..N}:
  LS[j,n] + BigM * (1-As[j,n]) >= LR[i,m] - BigM * (1-t[i,j]) - BigM * (1-Ar[i,m]);

subject to c19{i in 1..R, j in 1..R, m in 1..M, n in 1..M : i<>j && m<>n}:
  DR[i,m] >= DR[j,n] - BigM * p[i,j] - BigM * (1-Ar[i,m]) - BigM * (1-Ar[j,n]);

subject to c20{i in 1..S, j in 1..S, m in 1..N, n in 1..N : i<>j && m<>n}:
DS[i,m] >= DS[j,n] - BigM * q[i,j] - BigM * (1-As[i,m]) -
BigM * (1-As[j,n]);

Sample data file (Modeldata.txt)

param P:=3;
param R:=3;
param S:=2;
param M:=2;
param N:=2;

param loaded (tr):= 1 2 3:=
  1  50 50 100
  2  60 70 70
  3  20 30  0;

param needed (tr):= 1 2:=
  1 130  70
  2  60 140
  3  30  20;

param moving (tr):= 1 2:=
  1  20 40
  2  40 80;

param loading:= 1  2 2 4 3 2;
param delay:=0;
param BigM:= 100000;

Run file (Modelrun.txt)

reset;
option solver cplex;
option cplex_options "timing 1";
option show_stats 1;
option presolve 0;
model model.txt;
data modeldata.txt;
solve > modelout.txt;
display _varname > modelout.txt;
display _conname > modelout.txt;
display X > modelout.txt;
display Ar > modelout.txt;
display p > modelout.txt;
display As > modelout.txt;
display q > modelout.txt;
display DR > modelout.txt;
display LR > modelout.txt;
display DS > modelout.txt;
display LS > modelout.txt;
display t > modelout.txt;
expand > modelout.txt;
exit;
## Appendix B  Data sets used in the research

### Table B-1. Data set 1

Number of units of products loaded in receiving trucks and needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th>Shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>70</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>30</td>
</tr>
</tbody>
</table>

Distance between docks

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
</tbody>
</table>

Loading and unloading time for a unit of product

<table>
<thead>
<tr>
<th>Product</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
Table B-2. Data set 2

Number of units of products loaded in receiving trucks and needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th>Shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>87</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>52</td>
<td>93</td>
</tr>
<tr>
<td>3</td>
<td>105</td>
<td>83</td>
</tr>
</tbody>
</table>

Distance between docks

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
</tbody>
</table>

Loading and unloading time for a unit of product

<table>
<thead>
<tr>
<th>Product</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
Table B-3. Data set 3

Number of units of products loaded in receiving trucks and needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th>Shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>120</td>
<td>175</td>
</tr>
<tr>
<td>2</td>
<td>78</td>
<td>93</td>
</tr>
<tr>
<td>3</td>
<td>37</td>
<td>46</td>
</tr>
</tbody>
</table>

Distance between docks

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
</tbody>
</table>

Loading and unloading time for a unit of product

<table>
<thead>
<tr>
<th>Product</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
Table B-4. Data set 4

Number of units of products loaded in receiving trucks and needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th>Shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>140</td>
<td>37</td>
</tr>
<tr>
<td>2</td>
<td>117</td>
<td>80</td>
</tr>
</tbody>
</table>

Distance between docks

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
</tbody>
</table>

Loading and unloading time for a unit of product

<table>
<thead>
<tr>
<th>Product</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
**Table B-5. Data set 5**

Number of units of products loaded in receiving trucks and needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th>Shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>203</td>
<td>111</td>
</tr>
<tr>
<td>2</td>
<td>22</td>
<td>144</td>
</tr>
<tr>
<td>3</td>
<td>64</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>65</td>
<td>105</td>
</tr>
</tbody>
</table>

Distance between docks

<table>
<thead>
<tr>
<th>Shipping dock</th>
<th>Receiving dock</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
</tr>
</tbody>
</table>

Loading and unloading time for a unit of product

<table>
<thead>
<tr>
<th>Product</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
#### Table B-6. Distances between receiving and shipping docks used for data sets 6 to 10.

<table>
<thead>
<tr>
<th>Receiving dock</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
<td>130</td>
<td>150</td>
<td>170</td>
<td>190</td>
<td>210</td>
</tr>
<tr>
<td>2</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
<td>130</td>
<td>150</td>
<td>170</td>
<td>190</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
<td>130</td>
<td>150</td>
<td>170</td>
</tr>
<tr>
<td>4</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
<td>130</td>
<td>150</td>
</tr>
<tr>
<td>5</td>
<td>130</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
<td>130</td>
</tr>
<tr>
<td>6</td>
<td>150</td>
<td>130</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
<td>110</td>
</tr>
<tr>
<td>7</td>
<td>170</td>
<td>150</td>
<td>130</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
<td>90</td>
</tr>
<tr>
<td>8</td>
<td>190</td>
<td>170</td>
<td>150</td>
<td>130</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
<td>70</td>
</tr>
<tr>
<td>9</td>
<td>210</td>
<td>190</td>
<td>170</td>
<td>150</td>
<td>130</td>
<td>110</td>
<td>90</td>
<td>70</td>
<td>50</td>
</tr>
</tbody>
</table>

#### Table B-7. Loading and unloading time for products used for data sets 6 to 10.

<table>
<thead>
<tr>
<th>Product</th>
<th>Loading time</th>
<th>Product</th>
<th>Loading time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>11</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>12</td>
<td>1</td>
</tr>
</tbody>
</table>
Table B-8. Data set 6

Number of units of products loaded in receiving trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving truck</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>125</td>
<td>45</td>
<td>0</td>
<td>183</td>
<td>84</td>
<td>46</td>
<td>91</td>
</tr>
<tr>
<td>2</td>
<td>35</td>
<td>134</td>
<td>187</td>
<td>32</td>
<td>105</td>
<td>26</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>113</td>
<td>76</td>
<td>34</td>
<td>105</td>
<td>80</td>
<td>86</td>
</tr>
<tr>
<td>4</td>
<td>45</td>
<td>0</td>
<td>116</td>
<td>2</td>
<td>114</td>
<td>122</td>
<td>106</td>
</tr>
<tr>
<td>5</td>
<td>62</td>
<td>129</td>
<td>118</td>
<td>56</td>
<td>0</td>
<td>0</td>
<td>144</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>142</td>
<td>54</td>
<td>78</td>
<td>115</td>
<td>8</td>
<td>88</td>
</tr>
<tr>
<td>7</td>
<td>70</td>
<td>44</td>
<td>93</td>
<td>0</td>
<td>41</td>
<td>124</td>
<td>123</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>136</td>
<td>34</td>
<td>104</td>
<td>0</td>
<td>147</td>
<td>0</td>
</tr>
</tbody>
</table>

Number of units of products needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Shipping truck</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>1</td>
<td>221</td>
<td>54</td>
<td>135</td>
<td>48</td>
<td>0</td>
<td>116</td>
</tr>
<tr>
<td>2</td>
<td>98</td>
<td>178</td>
<td>66</td>
<td>78</td>
<td>200</td>
<td>33</td>
</tr>
<tr>
<td>3</td>
<td>54</td>
<td>0</td>
<td>160</td>
<td>146</td>
<td>76</td>
<td>101</td>
</tr>
<tr>
<td>4</td>
<td>92</td>
<td>146</td>
<td>32</td>
<td>88</td>
<td>81</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>111</td>
<td>126</td>
<td>201</td>
<td>0</td>
<td>98</td>
<td>103</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>167</td>
<td>0</td>
<td>213</td>
<td>63</td>
<td>50</td>
</tr>
<tr>
<td>7</td>
<td>187</td>
<td>43</td>
<td>158</td>
<td>38</td>
<td>60</td>
<td>97</td>
</tr>
<tr>
<td>8</td>
<td>77</td>
<td>129</td>
<td>76</td>
<td>89</td>
<td>86</td>
<td>0</td>
</tr>
</tbody>
</table>
Table B-9. Data set 7

Number of units of products loaded in receiving trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>86</td>
<td>43</td>
<td>45</td>
<td>116</td>
<td>2</td>
<td>114</td>
<td>122</td>
<td>106</td>
<td>6</td>
<td>62</td>
</tr>
<tr>
<td>2</td>
<td>105</td>
<td>0</td>
<td>80</td>
<td>86</td>
<td>43</td>
<td>45</td>
<td>0</td>
<td>116</td>
<td>114</td>
<td>122</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>80</td>
<td>86</td>
<td>43</td>
<td>45</td>
<td>116</td>
<td>0</td>
<td>128</td>
<td>143</td>
<td>106</td>
</tr>
<tr>
<td>4</td>
<td>86</td>
<td>132</td>
<td>73</td>
<td>0</td>
<td>167</td>
<td>26</td>
<td>192</td>
<td>48</td>
<td>41</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>146</td>
<td>67</td>
<td>134</td>
<td>98</td>
<td>45</td>
<td>0</td>
<td>76</td>
<td>0</td>
<td>67</td>
<td>44</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>130</td>
<td>0</td>
<td>47</td>
<td>86</td>
<td>143</td>
<td>35</td>
<td>75</td>
<td>46</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>134</td>
<td>80</td>
<td>86</td>
<td>43</td>
<td>145</td>
<td>116</td>
<td>27</td>
<td>114</td>
<td>122</td>
<td>55</td>
</tr>
</tbody>
</table>

Number of units of products needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>105</td>
<td>80</td>
<td>86</td>
<td>42</td>
<td>45</td>
<td>116</td>
<td>0</td>
<td>122</td>
<td>106</td>
</tr>
<tr>
<td>2</td>
<td>106</td>
<td>0</td>
<td>62</td>
<td>129</td>
<td>118</td>
<td>56</td>
<td>144</td>
<td>96</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>62</td>
<td>129</td>
<td>59</td>
<td>56</td>
<td>177</td>
<td>110</td>
<td>55</td>
<td>99</td>
</tr>
<tr>
<td>4</td>
<td>75</td>
<td>156</td>
<td>24</td>
<td>197</td>
<td>65</td>
<td>54</td>
<td>33</td>
<td>29</td>
<td>132</td>
</tr>
<tr>
<td>5</td>
<td>112</td>
<td>99</td>
<td>0</td>
<td>0</td>
<td>146</td>
<td>89</td>
<td>7</td>
<td>156</td>
<td>68</td>
</tr>
<tr>
<td>6</td>
<td>70</td>
<td>44</td>
<td>93</td>
<td>179</td>
<td>0</td>
<td>0</td>
<td>153</td>
<td>0</td>
<td>123</td>
</tr>
<tr>
<td>7</td>
<td>69</td>
<td>100</td>
<td>129</td>
<td>118</td>
<td>56</td>
<td>144</td>
<td>130</td>
<td>34</td>
<td>142</td>
</tr>
</tbody>
</table>
Table B-10. Data set 8

Number of units of products loaded in receiving trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>3</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>150</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>400</td>
<td>0</td>
<td>150</td>
<td>0</td>
</tr>
</tbody>
</table>

Number of units of products needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>104</td>
<td>137</td>
<td>125</td>
<td>3</td>
<td>81</td>
<td>137</td>
<td>64</td>
<td>0</td>
<td>99</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>116</td>
<td>146</td>
<td>36</td>
<td>78</td>
<td>40</td>
<td>60</td>
<td>53</td>
<td>69</td>
<td>77</td>
<td>75</td>
</tr>
<tr>
<td>3</td>
<td>54</td>
<td>73</td>
<td>23</td>
<td>73</td>
<td>138</td>
<td>94</td>
<td>124</td>
<td>89</td>
<td>0</td>
<td>82</td>
</tr>
<tr>
<td>4</td>
<td>98</td>
<td>75</td>
<td>58</td>
<td>18</td>
<td>117</td>
<td>66</td>
<td>113</td>
<td>89</td>
<td>116</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>98</td>
<td>80</td>
<td>248</td>
<td>21</td>
<td>47</td>
<td>15</td>
<td>11</td>
<td>31</td>
<td>199</td>
</tr>
<tr>
<td>6</td>
<td>124</td>
<td>12</td>
<td>28</td>
<td>103</td>
<td>66</td>
<td>53</td>
<td>22</td>
<td>105</td>
<td>137</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>50</td>
<td>152</td>
<td>138</td>
<td>49</td>
<td>33</td>
<td>73</td>
<td>69</td>
<td>60</td>
<td>113</td>
<td>13</td>
</tr>
<tr>
<td>8</td>
<td>139</td>
<td>14</td>
<td>66</td>
<td>40</td>
<td>130</td>
<td>112</td>
<td>40</td>
<td>101</td>
<td>0</td>
<td>108</td>
</tr>
<tr>
<td>9</td>
<td>90</td>
<td>140</td>
<td>156</td>
<td>38</td>
<td>72</td>
<td>35</td>
<td>44</td>
<td>118</td>
<td>44</td>
<td>13</td>
</tr>
<tr>
<td>10</td>
<td>129</td>
<td>88</td>
<td>113</td>
<td>137</td>
<td>49</td>
<td>83</td>
<td>12</td>
<td>95</td>
<td>0</td>
<td>44</td>
</tr>
<tr>
<td>11</td>
<td>181</td>
<td>0</td>
<td>32</td>
<td>71</td>
<td>22</td>
<td>52</td>
<td>93</td>
<td>138</td>
<td>17</td>
<td>144</td>
</tr>
</tbody>
</table>
Table B-11. Data set 9

<table>
<thead>
<tr>
<th>Receiving truck</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>44</td>
<td>136</td>
<td>147</td>
<td>114</td>
<td>105</td>
<td>116</td>
<td>122</td>
<td>106</td>
<td>6</td>
<td>62</td>
<td>130</td>
<td>144</td>
<td>118</td>
<td>112</td>
<td>122</td>
</tr>
<tr>
<td>2</td>
<td>104</td>
<td>0</td>
<td>124</td>
<td>123</td>
<td>41</td>
<td>149</td>
<td>116</td>
<td>114</td>
<td>26</td>
<td>116</td>
<td>26</td>
<td>113</td>
<td>122</td>
<td>122</td>
<td>122</td>
</tr>
<tr>
<td>3</td>
<td>34</td>
<td>80</td>
<td>97</td>
<td>101</td>
<td>116</td>
<td>116</td>
<td>86</td>
<td>128</td>
<td>0</td>
<td>143</td>
<td>106</td>
<td>116</td>
<td>116</td>
<td>116</td>
<td>116</td>
</tr>
<tr>
<td>4</td>
<td>147</td>
<td>132</td>
<td>173</td>
<td>192</td>
<td>80</td>
<td>192</td>
<td>192</td>
<td>15</td>
<td>0</td>
<td>85</td>
<td>142</td>
<td>54</td>
<td>54</td>
<td>115</td>
<td>115</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shipping truck</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>1</td>
<td>119</td>
<td>80</td>
<td>86</td>
<td>166</td>
<td>116</td>
<td>0</td>
<td>122</td>
<td>106</td>
<td>137</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>143</td>
</tr>
<tr>
<td>2</td>
<td>170</td>
<td>0</td>
<td>142</td>
<td>129</td>
<td>124</td>
<td>60</td>
<td>124</td>
<td>96</td>
<td>117</td>
<td>106</td>
<td>89</td>
<td>127</td>
<td>84</td>
<td>84</td>
<td>84</td>
</tr>
<tr>
<td>3</td>
<td>36</td>
<td>58</td>
<td>129</td>
<td>57</td>
<td>101</td>
<td>67</td>
<td>75</td>
<td>83</td>
<td>68</td>
<td>68</td>
<td>68</td>
<td>68</td>
<td>68</td>
<td>68</td>
<td>68</td>
</tr>
<tr>
<td>4</td>
<td>146</td>
<td>54</td>
<td>24</td>
<td>146</td>
<td>137</td>
<td>53</td>
<td>137</td>
<td>60</td>
<td>132</td>
<td>69</td>
<td>29</td>
<td>48</td>
<td>12</td>
<td>87</td>
<td>87</td>
</tr>
</tbody>
</table>

85
Table B-12. Data set 10

Number of units of products loaded in receiving trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Receiving trucks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>4</td>
<td>114</td>
</tr>
<tr>
<td>5</td>
<td>40</td>
</tr>
</tbody>
</table>

Number of units of products needed by shipping trucks.

<table>
<thead>
<tr>
<th>Product</th>
<th>Shipping trucks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>70</td>
</tr>
</tbody>
</table>
Appendix C  Scheduling algorithm coded in visual basic programming language

**Declarations**

```vbnet
Option Base 1
Public RUNTIME As Integer
Public R, S, M, N, P As Integer
Public Loaded(100, 20) As Integer
Public Needed(100, 20) As Integer
Public TruckChange As Integer
Public Distance(100, 100) As Integer
Public DistanceRT(100), DistanceTS(100) As Integer
Public Loading(20) As Integer
Public SW As Integer
Public Transfer(100, 100, 20), TransferTemp(100, 100, 20) As Integer
Public FreeRdock, FreeSdock, TransCount As Integer
Public RLoaded(100, 20), SNeeded(100, 20) As Integer
Public RSequence(100), SSequence(100) As Integer
Public Rremain(100), Sremain(100) As Boolean
Public Storage(20) As Integer
Public StorageLtime, StorageIn, StorageOut As Integer
Public LastR, LastS, LastRseq, LastSseq As Integer
Public EnterR, EnterS As Integer
Public RDocking(100), SDocking(100) As Boolean
Public InDockR(100), InDockS(100) As Integer
Public RDockLeave(100), SDockLeave(100) As Integer

Type Truck
    DockNo As Integer
    DTime As Integer
    LTime As Integer
    LTimeFinal As Integer
End Type

Type TransRecord
    From As Integer
    To As Integer
    Quantity As Integer
    LoadingTime As Integer
    TransferTime As Integer
End Type

Public TransOrder(1000) As TransRecord
Public RT(100) As Truck
Public ST(100) As Truck
Public OperationTime As Integer

**Scheduling procedure – section 5.2.3**

Public Sub Schedule()
    Call Copy
    Call FindMaxTransfer
    Do
```
RUNTIME = RUNTIME + 1
a = 0
b = 0
For k = 1 To P
    a = a + RLoaded(LastR, k)
    b = b + SNeeded(LastS, k)
Next k
If a = 0 Then
    Call DeAssignRD(LastR)
End If
If b = 0 Then
    Call DeAssignSD(LastS)
End If
If a = 0 And b = 0 Then
    Call FindMaxTransfer
ElseIf a = 0 And b <> 0 Then
    Call FindMaxTransferTo(LastS)
ElseIf a <> 0 And b = 0 Then
    Call FindMaxTransferFrom(LastR)
ElseIf a <> 0 And b <> 0 Then
    If FreeRdock > 0 And FreeSdock > 0 Then
        Call FindMaxTransfer
    ElseIf FreeRdock > 0 Then
        Call FindMaxTransferTo(LastS)
    ElseIf FreeSdock > 0 Then
        Call FindMaxTransferFrom(LastR)
    Else
        Call FindMaxTransferInDock
    End If
End If
End If
Call PrintDockStatus

c = 0
d = 0
For i = 1 To R
    If Rremain(i) Then c = c + 1
Next i
For i = 1 To S
    If Sremain(i) Then d = d + 1
Next i
If c = 0 And d > 0 Then
    For i = 1 To S
        If Sremain(i) Then
            Call TransFromStorage(i, 0)
            d = d - 1
        End If
    Next i
End If
Loop Until c = 0 And d = 0
maxt = 0
a = 0
For j = 1 To S
    If ST(j).LTime >= maxt Then
        maxt = ST(j).LTime
        a = j
    End If
Next j
mint = 10000
For i = 1 To R
    If RT(i).DTime <= mint Then
        mint = RT(i).DTime
    End If
Next i
OperationTime = maxt - mint
Call Report
End Sub

**Procedure 1 (Update product transfer) – section 5.2.4**

Public Sub UpdateTransfer()
    For i = 1 To R
        For j = 1 To S
            For k = 1 To P
                a = SNeeded(j, k) - RLoaded(i, k)
                If a >= 0 Then
                    a = RLoaded(i, k)
                Else
                    a = SNeeded(j, k)
                End If
                TransferTemp(i, j, k) = a
            Next k
        Next j
    Next i
End Sub

**Procedure 2 (Finding a pair of unscheduled trucks) – section 5.2.5**

Public Sub FindMaxTransfer()
    Call UpdateTransfer
    maxt = 0
    For i = 1 To R
        If Rremain(i) Then
            For j = 1 To S
                If Sremain(j) Then
                    a = 0
                    For k = 1 To P
                        a = a + TransferTemp(i, j, k)
                    Next k
                    If a > maxt Then
                        maxt = a
                        LastR = i
                        LastS = j
                    End If
                End If
            Next j
        End If
    Next i
    If Not RDocking(LastR) Then
LastRseq = NextRSequence
RSequence(LastRseq) = LastR
Call AssignRD(LastR, FindRdock(LastR, LastS))
RDocking(LastR) = True
EnterR = EnterR + 1
End If
If Not SDocking(LastS) Then
LastSseq = NextSSequence
SSequence(LastSseq) = LastS
Call AssignSD(LastS, FindSdock(LastR, LastS))
SDocking(LastS) = True
EnterS = EnterS + 1
End If
If maxt > 0 Then
Call Trans(LastR, LastS)
End If
End Sub

Procedure 3(Finding a receiving truck to schedule) - section 5.2.6

Public Sub FindMaxTransferTo(ss)
Call UpdateTransfer
Check = False
For k = 1 To P
If Storage(k) >= SNeeded(ss, k) And SNeeded(ss, k) > 0 Then
Call TransFromStorage(ss, k)
Check = True
End If
Next k
If Check Then Exit Sub
maxt = 0
a = 0
t = LastR
For i = 1 To R
If Rremain(i) Then
a = 0
For k = 1 To P
a = a + TransferTemp(i, ss, k)
Next k
If a > maxt Then
maxt = a
LastR = i
End If
End If
Next i
If Not RDocking(LastR) Then
LastRseq = NextRSequence
RSequence(LastRseq) = LastR
Call AssignRD(LastR, FindRdock(LastR, LastS))
RDocking(LastR) = True
EnterR = EnterR + 1
End If
If t <> LastR And maxt > 0 Then
    Call Trans(LastR, LastS)
Else
    Call TransFromStorage(LastS, 0)
End If

End Sub

**Procedure 4(Finding a shipping truck to schedule) – section 5.2.7**

Public Sub FindMaxTransferFrom(rr)

    Call UpdateTransfer
    maxt = 0
    For j = 1 To S
        If Sremain(j) Then
            a = 0
            For k = 1 To P
                a = a + TransferTemp(rr, j, k)
            Next k
            If a > maxt Then
                maxt = a
                LastS = j
            End If
        End If
    Next j
    If Not SDocking(LastS) Then
        LastSseq = NextSSequence
        SSquence(LastSseq) = LastS
        Call AssignSD(LastS, FindSdock(LastR, LastS))
        SDocking(LastS) = True
        EnterS = EnterS + 1
    End If
    If maxt > 0 Then
        Call Trans(LastR, LastS)
    End If

End Sub

**Procedure 5(Finding product transfer between docking trucks) – section 5.2.8**

Public Sub FindMaxTransferInDock()

    z = False
    Call UpdateTransfer
    maxt = 0
    For i = 1 To M
        For j = 1 To N
            If InDockR(i) > 0 And InDockS(j) > 0 Then
                a = 0
                For k = 1 To P
                    a = a + TransferTemp(InDockR(i), InDockS(j), k)
Next k
If a > maxt Then
   maxt = a
   LastR = InDockR(i)
   LastS = InDockS(j)
   End If
End If
Next j
Next i

If maxt > 0 Then
   Call Trans(LastR, LastS)
   z = True
End If

If Not z Then
   For i = 1 To S
      If SDocking(i) Then
         zz = False
         For k = 1 To P
            If Storage(k) >= SNeeded(i, k) And SNeeded(i, k) > 0
               Call TransFromStorage(i, k)
               zz = True
            End If
         Next k
      End If
   Next i
   If Not zz Then
      mint = 10000
      For i = 1 To M
         a = 0
         For k = 1 To P
            a = a + RLoaded(InDockR(i), k)
         Next k
         If a < mint Then
            mint = a
            j = InDockR(i)
         End If
      Next i
      Call MoveToStorage(j)
      Call FindMaxTransferTo(LastS)
   End If
End If

End Sub

Procedure 6(Receiving dock assignment) – section 5.2.9

Public Function FindRdock(ByVal rnum As Integer, ByVal snum As Integer) As Integer
   FindRdock = 0
   If SW = 1 Then ' with dock door assignment
      mindist = 10000
      For i = M To 1 Step -1
         For j = N To 1 Step -1

If InDockR(i) = 0 Then
    If ST(snum).DockNo = 0 Then
        If InDockS(j) = 0 Then
            If Distance(i, j) <= mindist Then
                mindist = Distance(i, j)
                FindRdock = i
            End If
        End If
    Else
        If Distance(i, ST(snum).DockNo) <= mindist Then
            mindist = Distance(i, ST(snum).DockNo)
            FindRdock = i
        End If
    End If
End If

Next j

Next i

If SW = 2 Then  ' without dock door assignment
    mint = 10000
    For i = M To 1 Step -1
        If InDockR(i) = 0 Then
            If RDockLeave(i) <= mint Then
                mint = RDockLeave(i)
                FindRdock = i
            End If
        End If
    End If
    Next i
End If

End Function

Public Sub AssignRD(ByVal rnum As Integer, ByVal rdnum As Integer)
    assigned = 0
    FreeRdock = FreeRdock - 1
    If InDockR(rdnum) = 0 And assigned = 0 Then
        InDockR(rdnum) = rnum
        assigned = 1
        RT(rnum).DockNo = rdnum
        RT(rnum).DTime = RDockLeave(rdnum) + TruckChange
        RT(rnum).LTime = RT(rnum).DTime
    End If
End Sub

Public Sub DeAssignRD(ByVal rnum As Integer)
    If InDockR(RT(rnum).DockNo) = rnum Then
        FreeRdock = FreeRdock + 1
        InDockR(RT(rnum).DockNo) = 0
        RDockLeave(RT(rnum).DockNo) = RT(rnum).LTime
        Rremain(rnum) = False
    End If
End Sub
**Procedure 7 (Shipping dock assignment) - section 5.2.10**

Public Function FindSdock(ByVal rnum As Integer, ByVal snum As Integer) As Integer

    FindSdock = 0
    If SW = 1 Then ' with dock door assignment
        mindist = 10000
        If rnum > 0 Then
            For j = N To 1 Step -1
                For i = M To 1 Step -1
                    If InDockS(j) = 0 Then
                        If RT(rnum).DockNo = 0 Then
                            If InDockR(i) = 0 Then
                                If Distance(i, j) <= mindist Then
                                    mindist = Distance(i, j)
                                    FindSdock = j
                                End If
                            End If
                        Else
                            If Distance(RT(rnum).DockNo, j) <= mindist Then
                                mindist = Distance(RT(rnum).DockNo, j)
                                FindSdock = j
                            End If
                        End If
                    End If
                Next i
            Next j
        ElseIf rnum = -1 Then
            For j = N To 1 Step -1
                If InDockS(j) = 0 Then
                    If DistanceTS(j) <= mindist Then
                        mindist = DistanceTS(j)
                        FindSdock = j
                    End If
                End If
            Next j
        End If
    End If
    If SW = 2 Then ' without dock door assignment
        mint = 10000
        For i = N To 1 Step -1
            If InDockS(i) = 0 Then
                If SDockLeave(i) <= mint Then
                    mint = SDockLeave(i)
                    FindSdock = i
                End If
            End If
        Next i
    End If

End Function

Public Sub AssignSD(ByVal snum As Integer, ByVal sdnum As Integer)

    assigned = 0

End Sub
FreeSdock = FreeSdock - 1
If InDockS(sdnum) = 0 And assigned = 0 Then
    InDockS(sdnum) = snum
    assigned = 1
    ST(snum).DockNo = sdnum
    ST(snum).DTime = SDockLeave(sdnum) + TruckChange
    ST(snum).LTime = ST(snum).DTime
End If

End Sub

Public Sub DeAssignSD(ByVal snum As Integer)
    If InDockS(ST(snum).DockNo) = snum Then
        FreeSdock = FreeSdock + 1
        InDockS(ST(snum).DockNo) = 0
        SDockLeave(ST(snum).DockNo) = ST(snum).LTime
        Sremain(snum) = False
    End If

End Sub

**Procedure 8(Timing) - section 5.2.11**

Public Sub Timing()
    i = TransCount
    a = TransOrder(i).From
    b = TransOrder(i).To
    If a <> 0 And b <> 0 Then
        If RT(a).LTime >= ST(b).LTime - Distance(RT(a).DockNo, ST(b).DockNo) Then
            RT(a).LTime = RT(a).LTime + TransOrder(i).LoadingTime
            ST(b).LTime = RT(a).LTime + Distance(RT(a).DockNo, ST(b).DockNo)
        Else
            RT(a).LTime = ST(b).LTime - Distance(RT(a).DockNo, ST(b).DockNo) + TransOrder(i).LoadingTime
            ST(b).LTime = RT(a).LTime + Distance(RT(a).DockNo, ST(b).DockNo)
        End If
    End If
    If b = 0 Then
        If RT(a).LTime >= StorageLtime - DistanceRT(RT(a).DockNo) Then
            RT(a).LTime = RT(a).LTime + TransOrder(i).LoadingTime
            StorageLtime = RT(a).LTime + DistanceRT(RT(a).DockNo)
        Else
            RT(a).LTime = StorageLtime - DistanceRT(RT(a).DockNo) + TransOrder(i).LoadingTime
            StorageLtime = RT(a).LTime + DistanceRT(RT(a).DockNo)
        End If
    End If
    If a = 0 Then
        If StorageLtime >= ST(b).LTime - DistanceTS(ST(b).DockNo) Then
            StorageLtime = StorageLtime + TransOrder(i).LoadingTime
            ST(b).LTime = StorageLtime + DistanceTS(ST(b).DockNo)
        Else
Procedure 9 (Product transfer from temporary storage) – section 5.2.12

Public Sub TransFromStorage(ByVal snum As Integer, ByVal kk As Integer)
    a = 0
    b = 0
    If kk <> 0 Then
        Storage(kk) = Storage(kk) - SNeeded(snum, kk)
        StorageOut = StorageOut + SNeeded(snum, kk)
        a = a + SNeeded(snum, kk)
        b = b + SNeeded(snum, kk) * Loading(kk)
        SNeeded(snum, kk) = 0
    Else
        For k = 1 To P
            If Storage(k) >= SNeeded(snum, k) Then
                Storage(k) = Storage(k) - SNeeded(snum, k)
                StorageOut = StorageOut + SNeeded(snum, k)
                a = a + SNeeded(snum, k)
                b = b + SNeeded(snum, k) * Loading(kk)
                Output.List1.AddItem "Product type '" & k & "' transferred from temporary storage to shipping truck " & snum & " = " & SNeeded(snum, k)
                SNeeded(snum, k) = 0
            End If
        Next k
    End If
    Output.List1.AddItem ""
    TransCount = TransCount + 1
    With TransOrder(TransCount)
        .From = 0
        .To = snum
        .Quantity = a
        .LoadingTime = b
    End With
    Call Timing
    a = 0
    For k = 1 To P
        a = a + SNeeded(snum, k)
    Next k
    If a = 0 Then
DeAssignSD (snum)
End If

End Sub

Procedures used by other procedures

Public Sub Copy()
    For i = 1 To R
        For j = 1 To P
            RLoaded(i, j) = Loaded(i, j)
        Next j
    Next i
    For i = 1 To S
        For j = 1 To P
            SNeeded(i, j) = Needed(i, j)
        Next j
    Next i
End Sub

Returns the number of last position in the receiving sequence

Public Function NextRSequence()
    NextRSequence = 0
    For i = 1 To R
        If RSequence(i) = 0 And NextRSequence = 0 Then
            NextRSequence = i
        End If
    Next i
End Function

Returns the number of last position in the shipping sequence

Public Function NextSSequence()
    NextSSequence = 0
    For i = 1 To S
        If SSequence(i) = 0 And NextSSequence = 0 Then
            NextSSequence = i
        End If
    Next i
End Function

Transfers products from the receiving truck to the shipping truck

Public Sub Trans(rr, ss)
    a = 0
    b = 0
    For k = 1 To P
        Transfer(rr, ss, k) = TransferTemp(rr, ss, k)
public function load(lastR, k) = RLoaded(lastR, k) - Transfer(lastR, lastS, k)
sneeded(lastS, k) = SNeeded(lastS, k) - Transfer(lastR, lastS, k)
a = a + Transfer(rr, ss, k)
b = b + Transfer(rr, ss, k) * Loading(k)
Output.List1.AddItem "Product type '" & k & '" transferred from receiving truck '" & rr & " to shipping truck '" & ss & " = " & Transfer(rr, ss, k)

Next k
Output.List1.AddItem ""
TransCount = TransCount + 1
With TransOrder(TransCount)
  .From = rr
  .To = ss
  .Quantity = a
  .LoadingTime = b
End With
Call Timing

End Sub

Moves the products to the temporary storage

Public Sub MoveToStorage(ByVal rnum As Integer)
a = 0
b = 0
For k = 1 To P
  Storage(k) = Storage(k) + RLoaded(rnum, k)
  StorageIn = StorageIn + RLoaded(rnum, k)
a = a + RLoaded(rnum, k)
b = b + RLoaded(rnum, k) * Loading(k)
Output.List1.AddItem "Product type '" & k & '" transferred from receiving truck '" & rnum & " to temporary storage = " & RLoaded(rnum, k)
RLoaded(rnum, k) = 0
Next k
Output.List1.AddItem ""
TransCount = TransCount + 1
With TransOrder(TransCount)
  .From = rnum
  .To = 0
  .Quantity = a
  .LoadingTime = b
End With
Call Timing
DeAssignRD (rnum)

End Sub
Snapshots of the scheduling problem

Figure C-1. Scheduling program - snapshot (Main page)

Figure C-2. Scheduling program - snapshot (Data input page 1)
Figure C-3. Scheduling program - snapshot (Data input page 2)
Figure C-4. Scheduling program - snapshot (Data input page 3)

Figure C-5. Scheduling program - snapshot (Output page 1)
<table>
<thead>
<tr>
<th>Product type</th>
<th>Transferred from receiving truck</th>
<th>Transferred to shipping truck</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product type 1</td>
<td>31</td>
<td>20</td>
</tr>
<tr>
<td>Product type 2</td>
<td>32</td>
<td>20</td>
</tr>
<tr>
<td>Product type 3</td>
<td>33</td>
<td>20</td>
</tr>
<tr>
<td>Product type 4</td>
<td>34</td>
<td>20</td>
</tr>
<tr>
<td>Product type 5</td>
<td>35</td>
<td>20</td>
</tr>
</tbody>
</table>

Total number of products transferred from a receiving truck to a shipping truck:

- Total number of products transferred from receiving truck 1 to shipping truck 2 = 165
- Total number of products transferred from receiving truck 2 to shipping truck 3 = 150
- Total number of products transferred from receiving truck 3 to shipping truck 4 = 145
- Total number of products transferred from receiving truck 4 to shipping truck 5 = 140
- Total number of products transferred from receiving truck 5 to shipping truck 6 = 135

Figure C-6. Scheduling program - snapshot (Output page 2)
Figure C-7. Scheduling program - snapshot (Output page 3)