A SIMULATION APPROACH FOR THE PREDECTIVE CONTROL OF A
DISTRIBUTION CENTER

A Thesis in
Industrial Engineering
by
Lourdes A. Medina

© 2009 Lourdes A. Medina

Submitted in Partial Fulfillment
Of the Requirements
for the Degree of
Master of Science
August 2009
The thesis of Lourdes A. Medina was reviewed and approved* by the following:

Richard A. Wysk
Professor of Industrial and Manufacturing Engineering
Thesis Advisor

Vittaldas Prabhu
Professor of Industrial and Manufacturing Engineering

M. Jeya Chandra
Professor of Industrial and Manufacturing Engineering
Graduate Program Chair

*Signatures are on file in the Graduate School
Abstract

This thesis presents a Simulation for Predictive Control (SimPC) as a decision making tool for improving the operation of non-automated distribution centers (DCs). SimPC is focused on determining the viability of a given truck–dock assignment schedule, including arrival times and dock assignments for inbound and outbound trucks at a large DC. SimPC also serves as an evaluation tool to perform analysis of iterative control procedures of system parameter adjustments while searching for a viable operational schedule. SimPC can utilize real-time data from DC’s warehouse management system (WMS) to obtain the current state of the DC, which serves as the initial conditions for the simulation. SimPC emulates the decision rules imbedded in SAP WMS, which includes assigning tasks to system-guided resources, selecting storage locations for inbound operations and determining retrieval locations for outbound operations. The SimPC model provides insight for identifying critical scheduling components along with guidance for operational and tactical solutions. It also serves as a tool to verify these solutions.
Table of Contents

List of Tables ..................................................................................................................... vi
List of Figures ................................................................................................................... vii
Acknowledgements ............................................................................................................ ix

Chapter 1: Introduction and Research Objectives .............................................................. 1

Chapter 2: Literature Review .............................................................................................. 5
  2.1 Introduction ............................................................................................................... 5
  2.2 Warehouse Characterization ..................................................................................... 7
  2.3 Storage/Retrieval Resources ................................................................................... 10
  2.4 Warehouse Control Systems ................................................................................... 16
    2.4.1 Inventory Management ..................................................................................... 17
    2.4.2 System Simulation ............................................................................................ 19
  2.5 Decision Making ..................................................................................................... 20
    2.5.1 Strategic Decisions ........................................................................................... 22
    2.5.2 Tactical Decisions ............................................................................................. 23
    2.5.3 Operational Decisions ....................................................................................... 24
  2.6 Chapter Summary and Conclusions ........................................................................ 25

Chapter 3: DC Control ...................................................................................................... 27
  3.1 Introduction ............................................................................................................. 27
  3.2 DC Control .............................................................................................................. 28
  3.3 Decision Algorithms ............................................................................................... 35
    3.3.1 TRM .................................................................................................................. 36
    3.3.2 Put-away ........................................................................................................... 39
    3.3.3 Picking .............................................................................................................. 40
List of Tables

Table 1. Warehouse processes ............................................................................................ 8
Table 2. Warehouse resources ........................................................................................... 9
Table 3. Warehouse organizational policies ..................................................................... 10
Table 4. TRM versus LES ................................................................................................ 33
Table 5. State graph nodes ............................................................................................... 43
Table 6. State graph transitions ....................................................................................... 43
Table 7. Descriptive algebra ............................................................................................ 44
Table 8. Legend for truck-dock schedule comparison ...................................................... 63
Table 9. Data sets for allowance calculation ................................................................. 71
Table 10. Minitab’s Power and Sample Size for 1-Sample t test for the forklift allowances ......................................................................................................................... 73
Table 11. Results with constant inbound dock ............................................................... 75
Table 12. Results with constant outbound dock ............................................................. 75
Table 13. TRM weights experiment scenarios ............................................................... 80
List of Figures

Figure 1 Example of company structure ................................................................. 6
Figure 2. A generic supply chain network ............................................................... 6
Figure 3. The strategic level .................................................................................... 21
Figure 4. The tactical level ...................................................................................... 21
Figure 5. The operational level .............................................................................. 22
Figure 6. Single-deep rack ..................................................................................... 28
Figure 7. Drive-in racks .......................................................................................... 29
Figure 8. DC arrangement ..................................................................................... 30
Figure 9. SAP versus WMS .................................................................................... 32
Figure 10. Yard activities ....................................................................................... 33
Figure 11. TRM-WMS ............................................................................................ 34
Figure 12. Task filtration ....................................................................................... 36
Figure 13. TRM Priority Model ............................................................................. 38
Figure 14. Put-away Algorithm procedure for complete pallets ......................... 39
Figure 15. Search order for more than a complete pallet ...................................... 41
Figure 16. Search order for less than a complete pallet ........................................ 41
Figure 17. DC State Graph .................................................................................... 43
Figure 18. Inbound operations ............................................................................. 45
Figure 19. Forklift operations .............................................................................. 45
Figure 20. Pallet jack operations ......................................................................... 46
Figure 21. Retrieval to truck loading process for accumulation at the dock .......... 47
Figure 22. Retrieval to truck loading process for accumulation at the buffer area .. 48
Figure 23. Current control schema ..................................................................... 51
Figure 24. Current control schema and traditional simulation .............................. 52
Figure 25. Proposed control schema 1 ................................................................. 53
Figure 26. Proposed control schema 2 ................................................................. 54
I would like to express my sincere gratitude to my thesis advisor, Dr. Richard A. Wysk, for his guidance and feedback throughout the course of this research. I also thank Dr. Vittaldas Prabhu and R. Ravi Ravindran for their assistance, encouragement and valuable advice during my research work.

My deepest gratitude goes to my husband, Josué R. Crespo, and my parents, Pedro L. Medina and Ana A. Aviles, for their unconditional love and support in this venture. This work is dedicated to them.
Chapter 1

Introduction and Research Objectives

Discrete event simulation has been a widely used method to represent and evaluate engineering processes, and still represents an on-going area of research and development (Mansharamani, 1997). Traditionally, simulation models have been used as an analysis tool for complex systems so that the systems do not have to be built or prototyped in order to capture the dynamics and stochastic characteristics of complex systems and their interactions. Simulation models offer a higher resolution for systems’ analysis than other modeling techniques such as optimization and queuing models. This is the case for warehouses and distribution centers (DCs) in which simulation modeling has been essential in the analysis of layout arrangements, storage allocation strategies, and for the justification of equipment investment, among others.

DCs represent a fundamental part of many supply chains. They serve as warehouses for finished goods, but more importantly represent an essential link between manufacturers and retailers. While warehouses are primarily focused on storing finished goods, DCs have a broader scope of purpose. One such purpose is to continuously fulfill orders throughout the day, in order to minimize the time to get goods to a customer and facilitate a material flow from factories to customers. This requires complying with a specific schedule for the delivery of goods, which is the main motivation for this thesis.
Even though the majority of the research seems to focus on automated warehouses and distribution centers, manually operated facilities are still quite popular due to the difference in investment and flexibility provided by having forklift trucks as the main equipment utilized for material flow. Manual DCs are labor-intensive and most of the work is performed by fork truck operators. Even though the warehouse management system (WMS) assists in the control of inventory, orders and dispatching of resources, there is little visibility from WMS into operations. When there are multiple operations occurring simultaneously (i.e., storage/retrieval locations, task assignment policies), the lack of visibility adds variability to the system, making it difficult to guarantee the viability of a schedule for specific requirements. This leads to the use of simulation modeling, which has been identified as a powerful approach for the analysis and improvement of such complex systems (Banks, 1998). However, most of the simulation studies performed for non-automated DCs have focused in the traditional use of simulation as an evaluation tool to decide between design alternatives, and not for the pro-active control of the daily operations.

The study by Hara et al. (2003) is one of the few studies that addressed the use of online scheduling of forklift trucks in non-automated warehouses with the objective of improving forklift task planning. Other research in non-automated warehouses include the identification of improvements by the evaluation of WMSs (Gale, Oliveros, and Silvan, 2002), and the development of conceptual models of the warehouse operations (Zhou, Setavoraphan, and Chen, 2005).

The literature of non-automated DC management lacks simulation based control models, as contrasted with the extensive number of control-related studies in automated
DCs. Such differences in the use of simulation-based control models between automated and non-automated DCs may be explained by the differences in DCs properties. In automated warehouses there is little or no randomness and variability in the operations, given that the tasks are performed automatically in a deterministic manner. However, for non-automated warehouses the human factor is present, having multiple operators performing their work differently, which adds human variability. Furthermore, building simulation models for non-automated DCs or warehouses has been recognized as a difficult task, compared to automated DCs (Gale, Oliveros, and Silvan, 2002; Takakuwa et al., 2000). Takakuwa et al. (2000) and Zhou, Setavoraphan, and Chen (2005) provide a variety of guidelines for modeling non-automated DCs and warehouses.

The main objective of this thesis is to present a Simulation for Predictive Control (SimPC) as a tool to improve decision making for non-automated DCs to determine and improve the viability of a given truck-dock assignment schedule.

The development of SimPC for non-automated DCs requires a realistic representation of the operations, which depends on having a variety of conditions such as allowing real-time data management, following the same decision making procedures and measuring human variability as part of the stochastic features. An existent non-automated and highly complex DC from a Multi-national Company is used as an example in the development of the SimPC approach.

The organization of this thesis is as follows: Chapter 2 presents the warehousing literature with emphasis on the characterization of the warehouse, storage/retrieval resources, Warehouse Control Systems (WCS) and decision making. Chapter 3 provides an overview of the DC control features incorporated in the development of SimPC,
including discussions about DC control, decision algorithms, and the development of a state graph. Chapter 4 goes in detail with the SimPC approach, demonstrating all of its elements, while illustrating its benefits in terms of constraint-based management. Chapter 5 is dedicated to the analysis needed for SimPC, including validation procedures and the definition of controllable variables and performance measures. Finally, conclusions and a discussion of future work directions are presented in Chapter 6.
Chapter 2

Literature Review

2.1 Introduction

Warehouses are important from a company structure (Figure 1) and a supply chain (Figure 2) standpoint. They represent one module of the system within the logistics control of the company, keeping interactions with the purchase module through customer orders and production as it feeds the warehouse inventory. When analyzing the supply chain perspective, its impact into the business is evident by serving as the connection between the company and the customer. This is a valuable aspect on most systems and is the focus of the work performed in this thesis. That is, this work looks at the implications of DC control for implications to the way customers are served.

The research on warehousing systems started gaining interest in the 1970s when the industry management efforts shifted from the improvement of productivity to the reduction of inventory (van den Berg, 1999). Since then, the literature concerning warehouses has increased to now covering a broad variety of topics. However, many of the studies deal with very isolated sub-problems, not necessarily looking at the big picture of warehouse operations (Rouwenhorst, 2000). This literature review includes the most relevant topics for this thesis: understanding the characterization of the warehouse operations, the review of storage/retrieval resources which have an impact in the
warehouse performance, a discussion about Warehouse Control Systems (WCS) that serves as the source of information for control, and the decision making involved in the warehouse operations.

![Diagram of company structure](image1)

**Figure 1.** Example of company structure (Draijer and Schek, 2004).

![Diagram of supply chain network](image2)

**Figure 2.** A generic supply chain network (Melo, Nickel and Saldanha-da-Gama, 2009).
2.2 Warehouse Characterization

Rouwenhorst et al. (2000) defined two different types of warehouses as: the distribution warehouse and the production warehouse. Both are aimed for storage, however, the first one, usually called a distribution center, and have the purpose of fulfilling customer orders, which entails a significant role from a supply chain perspective. On the other hand, production warehouses are used mainly for the storage of raw material, work-in-process and finished goods, being directly associated with manufacturing and assembly processes.

The two types of warehouses also differ in terms of the performance measures in which these are evaluated, aiming for maximum throughput in distribution warehouses and the best utilization of storage capacity for production warehouses (Rouwenhorst et al., 2000). This thesis is more focus on distribution warehouses based on the impact the proposed tool has from a customer stand-point; however, many of the concepts found in the literature apply for the two types of warehouses.

In general, warehouse operations can be summarized in terms of their processes, resources and organization. Processes, described in detail in Table 1, are required to accomplish the movement of products through different steps such that products can be stored and retrieved as needed. Different types of resources (Table 2), such as equipment and personnel, are necessary to operate the warehouse by performing the proper variety of processes.

In terms of the warehouse organization, there are many planning and control procedures that are necessary. Some of these decisions may have a direct impact in the system performance, as is the case of determining the process flow, which is performed at
the design stages. Organizational policies are required to control many of the processes, included in Table 3. The operator and equipment assignment policies, included at the end of the table, are also directly related to the storage/retrieval resources discussed in Section 2.3. Many of these policies may be included as part of the Warehouse Control System described in Section 2.4. Further, these include many of the decision making elements necessary at every warehouse, from the perspective of strategic, tactical and operational decisions in Section 2.5.

Table 1. Warehouse processes (Rouwenhorst et al., 2000).

<table>
<thead>
<tr>
<th>Process</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receiving</td>
<td>Process performed upon the arrival of products into the warehouse. May include the verification and repackaging of products.</td>
</tr>
<tr>
<td>Storage</td>
<td>Process of placing products into storage locations. May include two types of storage: a reserve area (stored in an economic way) and a forward area (stored for easy retrieval).</td>
</tr>
<tr>
<td>Replenishment</td>
<td>Material movement from the reserve area to the forward area.</td>
</tr>
<tr>
<td>Order picking</td>
<td>Retrieval of material from their storage location.</td>
</tr>
<tr>
<td>Sorting/Consolidation</td>
<td>Grouping of items destined for the same customer.</td>
</tr>
<tr>
<td>Shipping</td>
<td>Process that includes order verification, packaging and loading into trucks, trains or any other carrier.</td>
</tr>
</tbody>
</table>
Table 2. Warehouse resources (Rouwenhorst et al., 2000).

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage unit</td>
<td>Unit in which products are stored. Examples are: pallets, carton boxes and plastic boxes.</td>
</tr>
<tr>
<td>Storage system</td>
<td>These are very diverse, including from simple shelves up to highly automated systems that may consist of cranes and conveyors, among others.</td>
</tr>
<tr>
<td>Pick equipment</td>
<td>Used for the retrieval of items. Reach trucks are usually used for this purposes.</td>
</tr>
<tr>
<td>Order pick auxiliaries</td>
<td>Serve as support in the order picking operations, as is the case of bar code scanners.</td>
</tr>
<tr>
<td>Computer system</td>
<td>Used for the control of the processes by a warehouse management system.</td>
</tr>
<tr>
<td>Material handling equipment</td>
<td>For the preparation of the retrieval items. Examples include: sorter systems, palletizers and truck loaders.</td>
</tr>
<tr>
<td>Personnel</td>
<td>Important resource that manage the operations, having an impact in the warehouse performance.</td>
</tr>
</tbody>
</table>
Table 3. Warehouse organizational policies (Rouwenhorst et al., 2000)

<table>
<thead>
<tr>
<th>Policy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment Policy</td>
<td>Used to control the receiving process, determining the allocation of trucks to docks.</td>
</tr>
<tr>
<td>Storage Policy</td>
<td>Manages the storage process by deciding storage locations for the transportation of products through the storage system. Examples: dedicated storage policy, random storage policy, class based storage policy, family grouping, etcetera.</td>
</tr>
<tr>
<td>Forward/reserve and replenishment policy</td>
<td>Necessary when having reserve and forward areas.</td>
</tr>
<tr>
<td>Zoning policy</td>
<td>Deal with the management of multiple picking zones that result from the division of the picking area. Example: parallel and sequential zoning</td>
</tr>
<tr>
<td>Picking policy</td>
<td>There are the options of picking the orders one by one (single order picking) and in batches (batch picking).</td>
</tr>
<tr>
<td>Sorting policy</td>
<td>Necessary when using the batch picking policy. Two sorting policies are: pick and sort, and sort while pick.</td>
</tr>
<tr>
<td>Routing policy</td>
<td>Defines the sequence of retrievals and the route for these.</td>
</tr>
<tr>
<td>Dwell point policy</td>
<td>To decide the location of idle order picking equipment.</td>
</tr>
<tr>
<td>Sorter lane assignment policy</td>
<td>Necessary for consolidation and sorting processes.</td>
</tr>
<tr>
<td>Dock assignment policy</td>
<td>The allocation of orders and trucks to docks at the shipping process.</td>
</tr>
<tr>
<td>Operator and equipment assignment policies</td>
<td>Allocation of tasks to personnel and equipment.</td>
</tr>
</tbody>
</table>

2.3 Storage/Retrieval Resources

There are volumes of literature addressing the use of Automated Storage and Retrieval Systems (AS/RS) in distribution centers (DCs), but most do not examine the use of forklifts, which are capable of the same, or similar, sequencing rules. For almost
fifty years now, AS/RS have been used in distribution centers and warehouses to efficiently store and manage inventory (Yin and Rau, 2006). However, an AS/RS can be costly and also may not give the flexibility that a particular warehouse or DC requires (Hara et al., 2003). Most studies have been conducted to assure the efficient utilization and schedule of AS/RS resources in order to justify such a big investment. However, for this reason and other flexibility issues, forklifts continue to be utilized.

Hara et al. (2003) present research to deal directly with the use of forklifts. This work represents an effort to provide some improvements for forklift motion planning in terms of storage and operations efficiency while having variations in item variety and load size. A storage control system and scheduling work arrangement system was developed using simulated annealing as the search algorithm, which provided a good solution and concluding in the need for refining the re-scheduling model. They also discuss the issue of using offline versus online scheduling for forklifts. They found that online scheduling, which uses an onboard messaging screen, seems to provide a better performance when comparing the two alternatives through a simulation model.

In terms of retrieval selection strategies, a number of rules were found, being mostly applied for AS/RS: FCFS, STT (shortest-total travel time), SDT (shortest due date), NN (nearest neighbor), RNN (relief nearest neighbor), and RN (random). These rules are self-explanatory, with the exception of RNN which corresponds to a combination of FCFS and NN. Using RNN, only one of these two rules will be applied at a time, depending on a pre-determined parameter that specifies the maximum number of requests in a queue to perform FCFS rule, while NN is used when the queue length is equal to or greater than the parameter.
Yin and Rau (2006) point out the use of FCFS, STT, and SDT—not to show which rule is best, but to provide a framework for evaluating different rules through multi pass-simulation. Using experimentation, they concluded that dynamic rules—meaning the use of different rules under different time periods—work better than single rules. They specifically investigated the combination of an interactive look-ahead analysis from multi-pass simulation with an intelligent search from GA, while also considering different single dock layout configurations. This dynamic approach outperformed the previous dispatching rules based on various performance measures.

Randhawa and Schroff (1995) evaluated the use of NN, RNN and RN. In terms of throughput, their primary criterion, they found RNN to be the best retrieval rule for AS/RS under different single dock layouts. Lee and Schaefer (1997) address the use of FCFS as a commonly used rule for the sequencing of storage and retrieval requests. However, they also state that FCFS may not be a reasonable rule for the retrieval task in an AS/RS since these requests are electronic messages that can be easily rescheduled. This fact shows that in order to apply effective retrieval rules for forklifts, there is a need to have electronic messaging.

The implementation of these retrieval rules may be achieved with two different approaches: the dynamic case and the static case (Yin and Rau, 2006; Randhawa and Shroff, 1995; Lee and Schaefer, 1997). The dynamic case, also known as dynamic sequencing, performs the rescheduling of tasks as new requests arrive, requiring continuous computation as new requests are added (Lee and Schaefer, 1997). On the other hand, the static case, also called block wave sequencing, consists of batch processing the different retrievals before considering any other incoming requests (Lee
and Schaefer (1997). Lee and Schaefer (1997) implemented both approaches with the intention of experimentation, coming up with interesting conclusions that the best results were achieved with the dynamic case. Randhawa and Shroff (1995) and Yin and Rau (2006) selected a specific approach based on their understanding of which fits best the evaluated situation. Randhawa and Shroff (1995) preferred using the dynamic approach, stating that it “accounts for the dynamic nature of the system”. Yin and Rau (2006) chose the static approach with the argument that this will avoid the fact that with the other approach “a request with a due time will not be accomplished properly”.

In order to perform storage and retrieval tasks, either single cycles (SC) or dual cycles (DC) can be used. A single command cycle consists of either doing storage or retrieval, but not both (Randhawa and Shroff, 1995). A dual command cycle, also referred to as task interleaving, has both operations in one step which consists of storing a load from the inbound dock to a rack, picking up a load for retrieval, and then moving it to the outbound dock. The use of dual cycles has been studied in detail by Lee and Schaefer (1997), who formulated it as an assignment problem and solved it using the Hungarian method. The assignments correspond to forming the dual cycles by pairing storing with retrievals in order to minimize total travel distance. In addition, they also consider using a heuristic method developed from the assignment problem.

The use of genetic algorithms (GA) has been found to be a very popular approach to solve scheduling problems at distribution centers, especially for AS/RS. Li, Chen and Zhang (2006) developed a scheduling approach based on a GA to schedule five stacker cranes, having one per aisle, such that these could perform the assigned requests in the shortest possible time. Their results showed that their developed methodology
outperformed the conventional control. Wang et al. (2007) evaluate a similar problem, but applies the GA algorithm differently in which it result provides the group of rules, involving seven decisions, that are needed to achieve an optimal performance. In their formulation a chromosome represents a group of rules, which involved location and assignment decisions, among others.

Oliveira (2007) added additional complexity to this same stacker cranes problem by incorporating forklifts in charge of moving product from the crane’s input/out area to the docks and focusing on makespan as the performance measure. He compared the developed GA approach with simple rules (EDD, LRPT, SRPT, FCFS), showing the effectiveness of the developed method. Whitney et al. (1998) developed a GA heuristic considering different characteristics and objectives. In this case the order sequence is sensitive to the fixed production schedule, the contents of the inventory and the availability of the docks. The objective behind the implement algorithm was to minimize the time at the dock for trucks and railcars while also minimizing inventory. The GA results showed to be the best when compared with other methods, including heuristics and a search technique. Using one of these heuristics results as the initialization population improved even more the GA results.

Kim et al. (2003) used GA as part of a hybrid control architecture that included hierarchical and heterarchical features to solve the order picking problem in an automated system consisting of a conveyor and robots. This architecture meant having horizontal and vertical decisions capabilities, while making these decisions in real-time. Their approach was an adaptation of a control architecture initially designed for a general job shop scheduling and control environment. The results consisted of two hybrid versions
of their control architecture, showing no significant differences among these while both outperformed both the hierarchical and heterarchical approaches under different conditions. On the other hand, Watanabe, Furukawa and Watanabe (2001) looked at different distribution center problems, the stack palletizer schedule and the shipping sorter optimization problem, solving them with a GA approach. The solution of the first problem improved the efficiency almost 20% better than the conventional method, while the second problem results outperformed a random method.

Research was conducted by Rubrico et al. (2006) who followed an agent based approach to minimize the case picking time (makespan) for the different agents when performing the different routes that they had assigned by a previous process. Three procedures were defined to increase the spatial separation between agents and decrease the probability of interaction. The procedures, followed in this same order of preference are: no common sector (NCS) rule, free sector start (FSS) rule and least cost start (LCS) rule. The experimental phase was performed under different conditions of nodal distribution and item density, finding that in 37 out of the 40 experiments the proposed dispatching rule performed better than the solution with no dispatching rule.

In much of the literature, simulation modeling represents one of the mostly used tools for the evaluation and comparison of different techniques for the management of storage/retrieval resources. This brings up the paradigm of assuring that the model is a good representation of the studied operations. Takakuwa et al. (2000) to provide a procedure to build simulations models for handling activities of complicated and non-automated warehouses. Their proposed method, which was tested with an actual case, consists of two phases: the program for generating parameters of material handling, and
the simulation program. As part of their findings, they reinforce the complexity issue, by indicating that large-scale and non-automated DCs are usually more difficult to model than Automated Storage/Retrieval Systems (AS/RS), due to the replenishment operations at the non-automated facilities. Moreover, these scenarios are considered to be more difficult to analyze due to the variability and limitations of manual based work.

This thesis takes a realistic approach, by considering an existing DC operated by a Multi-national Company, where they use a sequencing rule. This rule does not correspond to any of the ones previously discussed as were found in the literature. The use of a specific warehouse control system (WCS) becomes the key, following the sequencing solution used by this system.

2.4 Warehouse Control Systems

The use of a Warehouse Control System (WCS) is vital for the management of complex and distribution centers (DCs) in real-time. Different kinds of software have been developed for these purposes offering a variety of capabilities, which include the fundamental feature of inventory tracking to more advanced capabilities as is the control of resources. Sophisticated solutions do not limit to just the management of information, by also including additional intelligence for decision making.

Nixon (1994) provides a good overview of the meaning of WCSs and serves as a guide to select the appropriate one based on the company needs. He reinforces the objective of these systems, which is to manage warehouse operations, and not just for locating, tracking and managing inventory. In this paper (Nixon, 1994), four features are identified as the main elements: radio frequency data communication (RFDC) devices,
dedicated localized computer hardware, automatic identification equipment and applications introduction software.

In the case of selecting the appropriate WCS, the following factors should be considered: system cost, system design, operational design, quotation completeness, implementation scheduling, training and documentation, related experience, financial strength, and support capability (Nixon, 1994). Nevertheless, some companies prefer developing their own solution. This is the case of Texas Instruments, which in 1977 developed and installed their first warehouse control system for an automated process (Bradshaw, 1987). At that time, the system objective was to control the conveyors, keep track of the storage locations for every item and monitor the inventory from the time it is received and placed in a pallet, to the time it is sent to manufacturing. Further, this paper provides the background information of the elements to consider in the development of WCSs, which include: hardware, software, debugging, system evolution, system upgrade, design considerations, installation considerations, simulation capabilities and requirements, etcetera.

The following Sub-sections elaborate on specific features found in the WCSs.

2.4.1 Inventory Management

Assuring inventory accuracy is a major element in the performance of a DC. This allows the creation of orders that can be satisfied effectively. Particularly, this facilitates complying with customer expectations while having an impact in the customer satisfaction measures commonly used to evaluate the performance of businesses.
Meanwhile, this also affects the operations of the business since poor inventory accuracy has implications in terms of the production schedules (Wayman, 1995).

WCSs serve as one of the elements necessary for inventory accuracy. These function as the tracking system managing a vast amount of information necessary for the control and management of inventory. Basic information that is necessary includes keeping track of quantities, state (e.g. usable, re-work, etc.) and location of the stock inventory. However, “information is power and the more information that can be controlled, the more powerful the more powerful the inventory system will become” (Wayman, 1995). For this reason, many tools include additional data such as buying information and vendor history, among others.

In addition to having a materials tracking system, Wayman (1995) defined other areas that have to work together for the improvement of inventory accuracy and control, which include the layout arrangement, training the personnel properly and having product identification. In terms of layout arrangement, the goal is to support easy access and efficient storage and retrieval. Safety is also a major issue when deciding the layout arrangement given that property damage can be caused depending on the traffic flow patterns, which further impacts the inventory accuracy of usable products.

Personnel training is important for the effective execution of the tracking system and the physically handling of the products (Wayman, 1995). The personnel is in charge of assuring the inventory accuracy not only by verifying the information in the tracking system, but by physically identifying and counting the products upon arrival and while stored. Policies as random verification of storage locations to compare with the information in the system further contribute to more inventory accuracy.
Proper product identification allows easily finding the products and satisfying the orders without additional delay. This includes having inventory clearly identified, properly stored, and secured. Verification and identification processes when the product is received assure proper labeling. Bar code technology is one of the tools facilitating this procedure, given that inventory accuracy is achieved by scanning the product and stocking location (Wayman, 1995). Radio-Frequency Identification (RFID) is expected to undertake this function in the near future, with more capabilities to assure inventory accuracy. Bowers et al. (2001) developed the inventory control system concept with RFID for a library application with similar capabilities as the ones needed in general for DCs.

Finally, “knowing where every item is and how much is on hand eliminates time spent searching, along with the associated costs” (Wayman, 1995).

2.4.2 System Simulation

A system simulation for WCSs does not refer to the actual simulation modeling we discuss in this thesis, but to an environment that allows testing and training with the software without making changes in the actual operation. This feature is used often for the validation of planned changes before these are incorporated into the actual operations.

Examples of the benefits of software simulation are shown by Draijer and Schenk (2004), who illustrate the use of the SAP simulation environments for educational purposes while teaching this ERP software. In the case of SAP, the simulation environment is a replicate of the operational environment with the information obtained at a specific time, not necessarily showing the current state of the operations.
The specific use of simulation for WCS software is discussed by Bradshaw (1987) while presenting a case study of Texas Instrument (TI). In this case they used a commercially available simulation that allowed them to construct an electronic model of the complete warehouse in detail, with the objective of validating their own WCS software. Constructing this simulation model involved four major steps: measuring the warehouse, entering data into tables, building separate zone models, linking zones together, and adding the AS/RS stacking program.

Some of the benefits TI found from having a simulation to test their WCS included: verifying all the hardware in the controller down to the last logic gate, testing at the home office rather than the warehouse, easier to correct control system errors and catch any early component failure, among others. A specific problem they were able to identify during this procedure included finding malfunctioning of the timing and communication between the boards.

Currently, some WCS include simulation capabilities as part of the software, as is the case the WCS of SAP, known as Warehouse Management System (WMS), which keeps a parallel testing environment (QL4) to the actual operations (PL4). In summary, there are cost benefits involved in these simulation capabilities, allowing system installation without warehouse shutdown which further avoids production losses (Bradshaw, 1987).

2.5 Decision Making

Multiple decisions are involved in the design, planning and daily management of warehousing and distribution operations. These have been classified in the literature in the context of three different levels, defined as strategic, tactical and operational
decisions. The decision types differ in terms of the impact time horizon, which include the long term, medium term and short term, respectively. Further, these are related between each other, having the strategic decisions affecting the tactical decisions which impact the operational decisions. Figures 3, 4 and 5 list a number of design problems at the strategic, tactical and operational levels, respectively, while being related to the context of the processes, resources and the organization.

Figure 3. The strategic level (Rouwenhorst, 2000).

Figure 4. The tactical level (Rouwenhorst, 2000).
2.5.1 Strategic Decisions

The literature includes models that have been developed for the solution of strategic, tactical and operational problems. At the strategic level, the publications found deal with the design of the process flow and the selection of warehouse systems based on economic considerations. Some of these studies include:

- the proposal of a systematic procedure for determining the size of a warehouse container (Roll, Rosenblatt and Kadosh, 1989),
- the development of a design algorithm for warehouses containing tote-sized loads (Dunkin, 1989), and
- the comparison in terms of performance between a standard ASRS with dual-shuttle ASRS through analytical expressions and simulation (Keserla and Peters, 1994).
2.5.2 Tactical Decisions

Tactical decisions mostly concern determining resource dimensions and the organization design (Rouwenhorst et al., 2000). Specifically, one of the common topics between various researchers refers to size and layout decisions. An optimization model is proposed to determine the optimal layout dimensions with the objective of minimizing handling distance, time, space utilization and cost (Berry, 1968; Bassan, Roll and Rosenblatt, 1980). Also, for conventional warehouses, simulation and analytical models have been used in the design procedure for size and layout decisions with concentration in storage capacity (Rosenblatt and Roll, 1984); and for the study of the layout effect in the response time (Pandit and Palekar, 1993).

Automated warehousing systems dominate the literature in regard to tactical decisions. Examples of these include:

- the proposal of a modeling method based on analytical expressions to maximize the throughput of a pick-to-belt order picking system (de Koster, 1996),

- simulation based studies to analyze the performance of sorting systems by the perturbation of variables such as the number of output lanes, order characteristics and control policies (Bozer and Sharp, 1985; Bozer, Quiroz and Sharp, 1988; Bozer, Quiroz and Sharp, 1991), and

- the presentation of case studies for the qualitative and quantitative comparison of pick-to-light equipment to alternate systems (Sharp et al., 1996).
2.5.3 Operational Decisions

The operational level literature can be grouped in five major categories: (1) batching, (2) storage policies, (3) routing and sequencing, (4) dwell point selection, and (5) storing and sequencing (Rouwenhorst, 2000). Batching algorithms have been evaluated through the aims of simulation (Elsayed and Stern, 1983) and analytical expressions (Elsayed and Unal, 1989). Particularly simulation has been used for algorithms developed for ASRS (Hwang and Lee, 1988) and pick-to-belt sorting systems (Armstrong, Cook and Saipe, 1979) and for sort-while-pick order picking operations (Gibson and Sharp, 1992).

In the evaluation of storage policies, the research goes from proposing heuristics to developing mathematical models. Jarvis and McDowell (1991) proposes a storage policy based on a heuristic for a conventional warehouse, while Muralidharan, Linn and Pandit (1995) uses simulation to evaluate three heuristics while analyzing the relocation of items in an ASRS. By the other hand, van den Berg and Sharp (1996) used linear programming for the product allocation of forward and reserve areas with the objectives of minimizing order picking costs and replenishment.

For routing and sequencing policies examples refer to Section 2.3. Limited studies have deal with the problem of dwell point selection, including:

- a liner programming based algorithm (Egbelu, 1991),
- the evaluation of policies through simulation modeling (Egbelu and Wu, 1993), and
- the response time analysis for multiple ASRS configurations and dwell point policies (Peters, Smith and Hale, 1996).
Storing and sequencing decisions have been commonly performed in the context of ASRS. Van Oudheusden et al. (1988) performed a simulation based case study for the storing and sequencing decision of a person-on-board ASRS. Schwartz et al. (1978) also did a simulation based study, but in this case to evaluate the throughput of ASRS under different storage and sequencing policies.

2.6 Chapter Summary and Conclusions

This chapter provided an overview of major warehousing concepts. Through the characterization of the warehousing operations it is described how these can be summarized in terms of processes, resources and organizational policies. Within this framework, major importance is given to the resources handling most of material movement, consisting of storage and retrievals in the accomplishment of inbound and outbound operations, respectively. Different models and algorithms have been developed for the management of these resources, being mostly focused on automated system.

Warehouse control systems are also described, not only for inventory management, but as a tool for the management and control of all the operations. These may incorporate a variety of capabilities, which are important to be aware in the consideration of flexibility and limitations these represent. A detail review on decision making further explores the warehousing literature in the context of strategic, tactical and operational decisions. Examples from the literature are included, finding the majority to be related to automated warehouses.

In conclusion, the warehousing literature is extensive, having from the study of specific problems to the broad analysis of warehouses management and control systems.
In general, most of the research has been focused on automated warehouses, which deterministic nature facilitates the application of complex rules. By the other hand, there is lack of developments for non-automated facilities that in general have the same requirements in terms of processes and organizational policies that are followed, while being different in terms of the resources performing these.
Chapter 3
DC Control

3.1 Introduction

The models developed in this research were created based on an operating distribution center (DC). This was a key element for the development of the Simulation for Predictive Control (SimPC) approach, by providing the necessary framework to define SimPC and its requirements in terms of data and structure.

This chapter provides an overview of the DC control features incorporated in the development of SimPC. A discussion of both generic and specific DC characteristics with details on storage locations, resources and the Warehouse Management System (WMS) is presented. This provides the structure of how the system is and can be controlled. One particularly valuable component for the DC that is modeled is WMS, which supports the real-time development of SimPC by providing status information on line and in real-time.

DC control algorithms are explained in detail. These algorithms are responsible of managing the fundamental procedures in the daily DC operations. Further, a state graph is developed to describe the level of detail necessary to capture the required fidelity of the DC processes, while demonstrating the granularity required for SimPC.
3.2 DC Control

The SimPC approach was developed based on a large-scale and stochastic, non-automated DC from a Multi-national Company with more than sixty thousand square meters, forty-five docks and about sixteen thousand storage locations. There are two different types of storage locations, including single-deep racks and drive-in racks.

The single-deep racks include six levels of storage spaces, being used for multiple purposes. As illustrated in Figure 6, the first level is used for the case picking operations having specific materials assigned to specific locations, while the second level stores the same material for replenishment. For instance, the materials are assigned to specific locations based on their weight, to avoid materials damage when mixing different materials by having a pre-determined picking route.

Figure 6. Single-deep rack
After case picking is performed there is no sorting of materials necessary, given that all the materials picked in one pallet correspond to the same shipment. The three higher levels of the single-deep racks are used primarily for the storage partial pallets (any material), while complete pallets may be stored if necessary when dealing with DC capacity problems, having drive-in racks (Figure 7) dedicated to their storage.

Figure 7. Drive-in racks (SJF Material Handling, 2009)

A complete pallet represents a specific number of cases that has been pre-defined for every material as the maximum amount that can be stored on a pallet. Partial pallets are considered to be incomplete, by not having an efficient utilization of the pallet space. For this reason, drive-in racks are used to exclusively store complete pallets. These are recommended when a big portion of the inventory shifts on a regularly basis, by allowing a much “denser arrangement” (Material Handling, 2009). Meanwhile, there is the constraint of having to store the same Stock Keeping Units (SKUs) on the same rack,
with a capacity between 18 and 30 pallets, depending on the pallet size. This further requires using intelligent rules, discussed in Section 3.3, to assure the efficient utilization of this space given that each drive-in rack is not dedicated to a specific material, where a class-based storage policy is employed to control these.

The class-based storage policy provides a general and organized approach for the storage of products, with the objective of grouping products of similar demand levels into the same class (Yin and Rau 2006). These classes are defined as A, B, and C (Figure 8); in which *Class A* represents the products with higher demand levels being stored in the nearest locations to the docks, while *Class C* refers to products with the lowest demand levels with farther placement in respect to the docks. This arrangement allows a large range of storage locations for materials in the same classification.

![Figure 8. DC arrangement](image)

In the case of resources, additional complexity is added by having shared resources, in which forklift trucks are considered as the most relevant ones, given that these work on almost any type of task (Figure 8), including replenishments, storage requests and retrieval tasks, at any point in time. Thus, forklift trucks are not tied to a specific inbound
material to be stored or outbound shipment to be fulfilled. Pallet jacks, which are mostly responsible for case picking operations, are shared among all shipments but limited to this type of task. As illustrated in Figure 8, these follow a specific order (S-shaped), to comply with the pallet weight arrangement. Moreover, other resources (e.g. loading teams, unloading teams, verifier operators) are also shared, but to a lesser extent. That is, these resources are shared with more than one truck, but have to work on these trucks one at a time. More details of the different resources and their significance in the DC processes are included in Section 3.4 in the form of a State Graph.

The latest version of Warehouse Control System (WCS) from SAP, known as WMS, serves as the management system for selecting the storage locations and the assignment of most of the resources described above, as well as many other functions. SAP, which stands for Systems, Applications and Products in Data Processing, is a well-known and widely used Enterprise Resource Planning (ERP) software product that integrates various business applications that update and process transactions in real-time mode. WMS, mostly designed for the DC scenario, works together with SAP in order to follow the ordering and shipping process from start to finish.

As illustrated in Figure 9, WMS functions as a decentralized stand alone system, by operating separated from the centrally operated SAP. Particularly, both SAP and WMS, were designed with capabilities to function with any other system (WCS and ERP, respectively), not necessarily an SAP product. In this case, the communication between these systems is enhanced by the Business Application Programming Interface (BAPI), which is the SAP standard interface. This allows SAP to control the ordering system following all the vendor and customer processes, while WMS manages the physical
picking and put-away of products in the DC. Further, it is important that these systems interact smoothly, provide flexibility for change adaptability and perform in a cost-efficient manner (SAP Library, 2006).

In addition to maintaining relevant information for inventory management, WMS includes two major functions known as: Yard Management (YM) and the Task and Resource Management (TRM). These come in the form of WMS modules, both being integrated to the SAP Logistics Execution System (LES). YM is integrated with LES through an interface, while TRM is fully integrated. The difference in responsibilities among TRM and LES are included in Table 4.
YM “extends warehouse management beyond the physical walls of the warehouse, enabling management and control from the time that goods arrive at the warehouse” (SAP Library, 2006). YM permits the creation of yard activities, as shown in Figure 10. This is considered outside the scope of this work, serving as a source of information, but having no impact on the tool development.

<table>
<thead>
<tr>
<th>Application</th>
<th>LES Role</th>
<th>TRM Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area of Responsibility</td>
<td>Movement, Operation demands, High-level scheduling and planning, Inventory management (down to storage bin level)</td>
<td>Distribution of work among warehouse resources, Control of execution pace, Reaction to execution events, Resource control, Control of intermediate storage bins and various bin characteristics, such as geographical constraints and optimization factors, Communication of work instructions to resources via their presentation devices</td>
</tr>
<tr>
<td>Information Exchange</td>
<td>Triggers processes in TRM</td>
<td>Requests guidance from LES, Feeds back data upon request completion</td>
</tr>
</tbody>
</table>

Figure 10. Yard activities
On the other hand, TRM has a big impact in SimPC development, as it deals with the internal DC operations. TRM (Figure 11) attempts to improve the efficiency of the DC processes by creating tasks, allocating these tasks to resources, controlling task execution, enabling resources to perform tasks via radio frequency devices and providing troubleshooting tools, among others (SAP Library, 2006). For troubleshooting, WMS holds a quality environment (QL), which is a copy of the production environment (PL), that allows for the testing of scenarios in TRM and the other functions within WMS. The TRM Monitor and TRM Monitor Alert are also useful in the troubleshooting procedures, while major function is to observe the operations and perform any manual adjustments (create tasks, assign tasks, change parameters, etc.).

Figure 11. TRM-WMS
For task execution, there is a Worker and Equipment Interface, having a relationship with every resource through an RF Terminal. The use of bar-codes with radio frequency devices facilitate this operation, making it more accurate. For the dynamic scheduling of the resources in real-time, it is necessary to have electronic task assignment capabilities, usually executed through the use of onboard messaging screens and handhelds.

For the specific allocation of tasks to resources, TRM has an imbedded algorithm that determines the task assignment based on a multiple-criteria analysis. This algorithm is explained in detail at Section 3.3. Also during the creation of tasks, other algorithms (also included in Section 3.3) not offered by WMS, are used in the decision of specific locations for the storage (Put-away Algorithm) and retrieval (Picking Algorithm) of materials. Both algorithms, developed for decision making at inbound and outbound operations, were programmed to work with WMS, making the decisions automatic for daily operations. WMS includes simple rules to be selected for these decisions, however, the ones actually used at the facility are more complex rules.

### 3.3 Decision Algorithms

For the continuous operation of the DC, three algorithms play a significant role for decisions required at any DC. These include: 1) who will perform the tasks (TRM Algorithm), 2) where to store the product at (Put-away Algorithm) and 3) where to retrieve the product from (Picking Algorithm). For instance, these are automated decision making tools working within WMS, requiring limited intervention and adding intelligence into the system with the intent of a better utilization of the resources.
3.3.1 TRM

The electronic task allocation to resources is an essential element of WMS’ TRM module, being limited to the control of forklift trucks and pallet jacks’ operations. When a resource becomes available, the system will assign the next task to performed, following the logic in Figure 12. From the list of all the “available” tasks, the ones “allowed” are selected for consideration. The tasks not being allowed may consist of two types: tasks that are never allowed depending on the specific resource (pallet jack vs. forklift truck), or tasks that are not allowed temporarily based on the state of the system.

![Figure 12. Task filtration (SAP Library, 2006)](image)

Having tasks that are never allowed based on the resource type, comply with the fact that forklift trucks are dedicated to tasks requiring material movement from different levels (e.g. movements from/to drive-in racks), while pallet jacks can only perform tasks in the floor level (e.g. case picking). Meanwhile the tasks that are not allowed temporarily based on the state of the system.
temporarily based on the state on the system, account for restrictions on the number of resources permitted simultaneously in specific zones of the DC. In this case, a task may not be included as “allowed” because either the origin or destination’s zone of the task has already reached its maximum number of resources permitted. Not only resources physically in the zone are considered when calculating the number of resources in the zones, but also the ones already assigned to work in that area.

From the group of allowed tasks, one task is selected for the available resource. For this TRM uses a priority model, called TRM Algorithm for the purpose of this work, which is based on a multiple-criteria analysis. Specifically, multiple calculations are performed, based on the different criteria and sub-criteria, in order to obtain a single priority value where the task with the highest value is selected. This calculation is time and case specific, requiring its dynamic calculation. The result varies based on the current location of the resources and the time elapsed since the task was created.

As illustrated in Figure 13, three main criteria are used for the task selection. These are: Latest starting date (LSD), Route and the Static and Synchronization. Weights are used to vary the importance of the different criteria in the final priority calculation, while each sub-criteria has also a weight for the priority calculation of its corresponding criteria. The LSD priority calculation requires system and task parameters, including task expiration time, task duration and speed of the slowest resource. LSD account with the necessity to comply with expiration dates, by increasing this priority as time elapses.
Various sub-criteria are involved with the Route priority calculation. Among the Route sub-criteria, only the “distance from source” element is variable since this value depends on the current resource location and the origin location for the task. The other sub-criteria elements for Route (Resource-handling unit, Resource-working area and Resource-type level) are resource dependent values that do not change with time, but still impact the final Route priority.

The Static and Synchronization criteria have two sub-criteria: the static priority and the synchronization priority. The static priority sub-criteria have two elements of its own, known as the activity and host sub-sub-criteria. These consist of pre-defined priority values that depend on the type of activity (inbound, outbound, replenishment, etc.) and the person that created the task, respectively. By the other hand, the synchronization sub-criteria only applies to related tasks depending on each other, being not employed in the evaluated DC.
3.3.2 Put-away

Even though the Put-away Algorithm decides the storage location for materials during the inbound operations, it was designed to improve the efficiency of the outbound operations. The travel distance requirement for the storage of materials is not considered, while the focus is on having a balanced distribution of material throughout the DC so as to minimize picking times. This is understandable from the perspective that storage locations later become retrieval location, having performance implications in the shipments completion process based on the DC size.

The Put-away Algorithm is an iterative process, being performed independently for each SKU when received in the inbound operations. As discussed earlier, the partial pallets received are stored in the three highest levels of the single deep racks, in which any location may be selected. However, complete pallets are stored following the balancing procedure, with Figure 14 as a general illustration, while complying with the class-based classification policy.

![Figure 14. Put-away Algorithm procedure for complete pallets](image)

As an initial step, the Put-away Algorithm determines if there are any replenishment requirements. This activity is designed specifically to ensure that the first and second level of the single deep racks, which correspond to the case picking and replenishment locations are available to service current customer requirements. This is only necessary
when that specific SKU is not available in the whole DC, given that replenishment tasks are created automatically when the material is available at the DC.

If no replenishment is necessary, the balancing calculation takes place, in which the DC can be seen as three DC’s in one. The objective is to maintain an equal amount of all the materials in each “sub-DC”. The amount of available material is evaluated in each of the three areas, and the area with the lowest amount is selected. After this, a specific drive-in rack is selected searching in a pre-determined order that depends of the material classification (ABC). In this process higher priority is given to the locations that have already in storage the same material SKU. All the pallets with the same SKU, analyzed within the same inbound truck, will be stored in the same selected drive-in rack as long as there is sufficient space.

After the selected location has been filled, if there are remaining pallets of that same material the analysis starts from the balance calculation phase, in which the calculation is made to decide in which area the remaining pallets should be stored. When finished with that material, a second material is selected to follow the same rules, and the process continues until all the materials have been assigned a storage location.

### 3.3.3 Picking

The Picking Algorithm deals with selecting the retrieval location for all the material requirements during the outbound operations. The intention of this algorithm is to minimize total pick times for each truck load by taking material from the nearest location to the truck dock while complying with two limitations for the specific case of drive-in racks. These limitations consist of a specific search order and complying with an
expiration date constraint. As part of the expiration analysis, the algorithm selects which pallet locations will comply with the expiration date constraint. For this, the oldest pallet’s expiration date is found to calculate an allowable time window. All the pallets fitting within that window will be evaluated selecting the nearest location to the dock as the retrieval location.

Similar to the Put-away, this algorithm is an iterative process where the decision is done per material, but in this case performing additional analysis based on the number of boxes per pallet. It is material per material and creates the tasks one by one. The amount of material cases a shipment requires, in contrast with the amount considered to be a complete pallet, govern the search order. If the total/remaining requirements for that material are more than a complete pallet, then the search order in Figure 15 is followed. However, if the total/remaining requirements for that material correspond to less than a complete pallet, the search order in Figure 16 is used.

After assigning a retrieval location for the evaluated material, the algorithm updates the number of cases remaining for the material and performs the same logic to find
retrieval locations for the remaining cases. For one same material, at the beginning it may consider only the rules for having more than a complete pallet (Figure 15). However, as the retrieval location assignments are performed, it may result in requirements of less than a complete pallet, using the logic in Figure 16.

For the materials requiring case picking, the selected locations are saved such that when all the materials have been evaluated, the case picking pallets can be formed. These pallets are arranged following a specific order based on the material weight, which is aligned with how the case picking is performed, as explained in Section 3.2. Specifically, heavy materials will be in the bottom of the pallet not damaging light-weight materials. In addition, a volume calculation is used to control the amount of cases to place in a pallet.

3.4 State Graph

A state graph was developed to further define and understand the procedures being followed at the DC. Most importantly, this shows that there is a generic set of tasks and resources associated with any DC, representing the foundation for the development of SimPC by delineating the level of detail needed to capture the fidelity of the operations. This provides a framework for this work, while serving as a contribution in the development of generalized models for the understanding and characterization of DC operations.

The graph, shown in Figure 17, provides a generic view of the possible states (nodes) and transitions (arrows) of the materials in the DC. The meaning of the nodes and transitions are provided in Tables 5 and 6, respectively; with descriptive algebra
included in Table 7 that further illustrates the transitions required to accomplish a specific state.

Figure 17. DC State Graph

Table 5. State graph nodes

<table>
<thead>
<tr>
<th>Node</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Inbound truck load (pallets)</td>
</tr>
<tr>
<td>B</td>
<td>Inbound truck load (cases)</td>
</tr>
<tr>
<td>Ci</td>
<td>Cases in bulk dock i</td>
</tr>
<tr>
<td>Di</td>
<td>Pallets at inbound dock i</td>
</tr>
<tr>
<td>Ej</td>
<td>Pallet in forklift j – TRM algorithm</td>
</tr>
<tr>
<td>Fl</td>
<td>Complete pallets in drive-in rack l</td>
</tr>
<tr>
<td>Gm</td>
<td>Partial pallets in single-deep rack m</td>
</tr>
<tr>
<td>Hn</td>
<td>Pallet at case picking rack n (level 2)</td>
</tr>
<tr>
<td>Ip</td>
<td>Pallet in buffer area p</td>
</tr>
<tr>
<td>Jk</td>
<td>Pallet at outbound dock k</td>
</tr>
<tr>
<td>Ko</td>
<td>Pallet at case picking rack o (level 1)</td>
</tr>
<tr>
<td>Mp</td>
<td>New pallet in buffer area p</td>
</tr>
<tr>
<td>Np</td>
<td>Wrapped pallet in buffer area p</td>
</tr>
<tr>
<td>Oq</td>
<td>Pallet in pallet jack q</td>
</tr>
<tr>
<td>P</td>
<td>Cases in truck load</td>
</tr>
</tbody>
</table>

Table 6. State graph transitions

<table>
<thead>
<tr>
<th>Transition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>Unload event</td>
</tr>
<tr>
<td>γ</td>
<td>Verify event</td>
</tr>
<tr>
<td>ε</td>
<td>Complete pallet event</td>
</tr>
<tr>
<td>η</td>
<td>PKO algorithm event</td>
</tr>
<tr>
<td>ν</td>
<td>Replenishment level 2 event</td>
</tr>
<tr>
<td>ρ</td>
<td>Drop on dock event</td>
</tr>
<tr>
<td>τ</td>
<td>Re-palletize event</td>
</tr>
<tr>
<td>φ</td>
<td>Request event</td>
</tr>
<tr>
<td>β</td>
<td>Palletize event</td>
</tr>
<tr>
<td>δ</td>
<td>PAO algorithm event</td>
</tr>
<tr>
<td>ζ</td>
<td>Partial pallet event</td>
</tr>
<tr>
<td>μ</td>
<td>Replenishment level 1 event</td>
</tr>
<tr>
<td>ξ</td>
<td>Damaged event</td>
</tr>
<tr>
<td>σ</td>
<td>Drop on buffer event</td>
</tr>
<tr>
<td>υ</td>
<td>Wrap event</td>
</tr>
<tr>
<td>ψ</td>
<td>Load event</td>
</tr>
</tbody>
</table>
Table 7. Descriptive algebra

<table>
<thead>
<tr>
<th>Expression</th>
<th>Algebraic Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F(B, \alpha) = C$</td>
<td></td>
</tr>
<tr>
<td>$F(A, \alpha) = F(C, \beta) = F(D, \gamma) = D$</td>
<td></td>
</tr>
<tr>
<td>$F(D_i, \delta) = F(F_i, \eta) = F(G_m, \eta) = F(H_n, \nu) = E_j$</td>
<td></td>
</tr>
<tr>
<td>$F(E_j, \varepsilon) = F(E_j, \zeta) = G_m$</td>
<td></td>
</tr>
<tr>
<td>$F(E_j, \mu) = H_n$</td>
<td></td>
</tr>
<tr>
<td>$F(E_j, \sigma) = F(O_q, \sigma) = I_p$</td>
<td></td>
</tr>
<tr>
<td>$F(E_j, \rho) = F(O_q, \rho) = F(J_k, \gamma) = J_k$</td>
<td></td>
</tr>
<tr>
<td>$F(E_j, \tau) = K_\delta$</td>
<td></td>
</tr>
<tr>
<td>$F(\zeta, \xi) = L$</td>
<td></td>
</tr>
<tr>
<td>$F(\zeta, \eta) = M_k$</td>
<td></td>
</tr>
<tr>
<td>$F(M_k, \psi) = N_k$</td>
<td></td>
</tr>
<tr>
<td>$F(N_k, \phi) = F(K_\delta, \eta) = O_i$</td>
<td></td>
</tr>
<tr>
<td>$F(\zeta, \psi) = P$</td>
<td></td>
</tr>
</tbody>
</table>

In the inbound operations (Figure 18), states A (pallets in truck) and B (cases in truck) represent the two types of inbound trucks being received at the DC with traditional rear load/unload and side load trucks. The unloading operation of a pallet by a forklift truck is represented through transition $\alpha$ that changes the state from a pallet in a truck (A) to a pallet in dock i ($D_i$). On the other hand, the material received in cases must be palletized, having an intermediate state of cases in dock i ($C_i$). An unloading team takes care of unloading the cases and organizing these into pallets for storage purposes. Also, within state $D_i$ there is a verification procedure represented by transition $\gamma$. This procedure is performed by one person in order to assure that the amount and type of material received agrees with the information in the system.
After the material has been verified, it is ready to be stored, where the storage tasks are created in WMS following the Put-away Algorithm that is used to determine the storage location. The resulting tasks may consist of cross-docking (state $J_k$—movement to outbound dock $k$), replenishment to case picking locations (state $K_o$—level 1 or state $H_n$—level 2), storage to a drive-in rack (state $F_l$) or storage to a single-deep rack (state $G_m$). All of the inbound related tasks are only assigned to forklift trucks (Figure 19), state represented by $E_i$, given that most of these tasks require the vertical movement capability that this equipment provides.
Beyond the inbound operations, forklift trucks are assigned all the replenishment tasks, which could represent the following state transitions: from F₁ to Hₙ, F₁ to K₀, or Hₙ to K₀. Each replenishment task is automatically generated by the system when the last material of a case picking location has been retrieved. As part of the outbound operations, forklift trucks are assigned to the movements from the drive-in racks (F₁) and single-deep racks (Gₘ) to either the outbound dock (Jₖ) or the buffer/wrapping area (Iₚ).

As shown, forklift trucks (Figure 19) are assigned to perform most tasks in the system (inbound, outbound, replenishment, internal movements, etc.) where the TRM algorithm assigns the next task to be performed. On the other hand, pallet jacks (state Oₚ and Figure 20), also controlled by TRM, are only involved in the outbound operations, performing case picking operations (serving as the intermediate state between states K₀ and Jₖ) and other floor pallet movements (state between states Nₚ and Jₖ).

Figure 20. Pallet jack operations
In terms of the outbound operations there are two types of accumulation, at the dock cages (Jk) or in a buffer area (Ip). In either accumulation type, the creation of the retrieval tasks in WMS is performed by the Picking Algorithm, which takes care of deciding the best locations to obtain the materials from. Both of these follow different processes in terms of states and transitions, having the corresponding state graphs segments illustrated through Figures 21 and 22, respectively. The first case is simple, requiring just two transitions, while in the physical system this represents loading case by case into the truck. The accumulation at the buffer area includes six transitions (palletize - M_p, wrapping - N_p and various movement processes), but a much faster loading of the truck as it is in pallets. The accumulation at the buffer is more requires the use of additional resources.

Figure 21. Retrieval to truck loading process for accumulation at the dock
3.5 Chapter Summary and Conclusions

This chapter outlined the major elements considered for the development of SimPC, while justifying the need for this tool. DCs have become more difficult to manage as a result of the high level of uncertainty caused by multiple variables and increased product variability (The number of different SKUs managed at a DC). Having shared resources seem to be a good practice in terms of flexibility and utilization, while this causes competition between the operational requirements, in which inbound and outbound trucks fight for the same set of resources. This may further have implications in the viability of the evaluated truck-dock assignment schedule.

The TRM Algorithm tries to deal with this problem by providing a set of rules to be followed when assigning tasks to system guided resources (mostly forklift trucks). However, this also introduces randomness that could make a difference in the viability of a specific schedule. This is particularly important from the observed relevance of forklift trucks into the operations, observed in the state graph since these are required for
many of the state transitions, being indispensable for the storage operations and any movement related to drive-in racks.

Having a dynamic scheduling approach has been identified as a good policy to be followed since it permits re-scheduling of the tasks as new tasks are added into the system (Hara et al. 2003). This is also supported by Randhawa and Shroff (1995) where the authors state that dynamic scheduling “accounts for the dynamic nature of the system”. Particularly, TRM provides a practical approach to the DC management as it considers many different aspects of the operation, when compared to the predominant focus of minimizing travel time. The use of additional criteria such as meeting specific deadlines seems to be more promising from a practical standpoint (van den Berg, 1999).

Additional variability is also obtained from the Put-away and Picking Algorithms, given that the decisions are case specific in which storages and retrieval locations fully depend on the current state of the DC. SimPC serves as the tool to deal with the DC uncertainty, serving as a traditional simulation analysis, as a real-time controller and a task generator, if necessary.
Chapter 4

SimPC Approach

4.1 Introduction

The intent of the simulation developed in this research (SimPC) is to model the resources within a DC and detail the interactions of the critical assignment of resources for the control of a DC, with a focus on critical decisions to be considered at any DC. This further serves as the framework for the application of SimPC as a traditional simulation analysis tool, or in the daily control of operations as part of the DC operational architecture.

This chapter describes in detail the SimPC approach. A general overview of the tool exemplifies the different scenarios where SimPC can be applied, from the traditional to forming part of the DC control, while also evaluating the fundamental DC elements incorporated into SimPC. Details about the model and data requirements delineate the features for implementation. Finally, the benefits that SimPC provides are discussed from a constraint-based DC management standpoint, proposing the use of SimPC as an iterative tool in the search for improved operation of resources and parameters that will guarantee the viability of a working plan.
4.2 Approach Overview

The current control schema for a DC is shown in Figure 23. SimPC can be used in conjunction with this schema in a variety of ways. Traditional simulation analysis (Figure 24) permits the evaluation of static control specifics by analyzing the performance of the DC for various control parameters and decisions. Particularly, control decisions can be examined in terms of their effect on the throughput of products and trucks to the DC. These evaluations could be of multiple types, from requiring simple parameters changes (e.g. number of resources) in the simulation code, to the necessity of major code modifications (e.g. different task assignment algorithm).

![Current control schema](image)

Figure 23. Current control schema
On the other hand, more benefits can be obtained if SimPC is incorporated within the control schema, having identified two potential arrangements. Figure 25 demonstrates the case in which SimPC interacts with the TRM Algorithm for the purpose of changing the algorithm’s parameters (weights, priorities, and etcetera) in order to develop control for a specific objective. To achieve this, prior traditional analysis should be performed as a benchmark to know the central operational points. This allows the perturbation of these parameters from the known central point, using SimPC as a real-time controller with the concepts of look-ahead and multi-pass simulation.
Figure 25. Proposed control schema 1

Figure 26 provides the second proposed arrangement, in which SimPC completely replaces the TRM module, by performing all of its responsibilities, from task creation to task assignment and monitoring. In addition to the decisions that this requires, SimPC works as a task generator, by sending tasks to the system guided resources, hence becoming the execution system for WMS. This further provides the flexibility of using different algorithms, beyond the TRM Algorithm, in the control of system guided resources. Complex rules, such as the ones found in the literature, may be considered for these purposes.
Using SimPC as a controller requires emulating the decisions rules followed in a daily basis as part of the DC operations. As Figure 27 illustrates, it has to incorporate a variety of inputs, controls and mechanisms in order to obtain the necessary outputs. The data input represents one of the biggest challenges from an implementation stand point, given that the best utilization of SimPC requires real-time data capabilities. Moreover, these inputs represent a major part in the development of this approach, since these correspond to the necessary information to run the simulation model and provide meaningful results.
The truck–dock assignment schedule plan is an essential part of this data input, providing information about inbound and outbound trucks. Additional details are included in Section 4.5. Within the scope of this work, it is assumed that this truck–dock assignment schedule plan is given, typically from an optimization model using time estimates, or some other rule-based methodology. The plan generated for either one of these optional methods will not be able to capture the dynamics of a large-scale and highly stochastic non-automated DC, sustaining the contribution of SimPC.

As noted in Figure 27, most of the data input consists of information providing the current state of the system, such that any analysis is made using the most realistic conditions. This information is obtained at the moment the simulation run is initiated. Some of these data elements include: current tasks for system guided resources, available resources and their state; information about the different inbound and outbound trucks,
the current stock, and various operational parameters, among others. The actual complexity, in terms of the scalability of this information, includes ranges like: a thousand simultaneous tasks, nine hundred materials and fifteen thousand possible storage/retrieval locations.

As part of the control elements, there are many operational parameters affecting the performance of the DC. These mainly consist of the major algorithms utilized for decision making: the TRM Algorithm, the Put-away Algorithm and the Picking Algorithm, all of which have been previously discussed. In addition, there are other control features, which extend to having a class based (A, B, and C) classification for the materials and storage locations, including capacity constraints in terms of the storage space and incorporating a dynamic capacity named aisle capacity to specify a maximum number of resources allowed simultaneously in specific areas.

Other parameters being considered are the mechanisms, which refer to the specific resources necessary to perform the different tasks. In terms of the outputs, traditional performance measures could be included, however in this diagram only some are included.

4.3 The Model

The simulation model consists of a Visual Basic (VB) software program which maintains information for discrete event simulation. The preference for VB, instead of a simulation package assures a fast simulation run of the model, which is required to use SimPC as a controller. Furthermore, the use of VB facilitates the incorporation of rule-based algorithms previously discussed. Still, the design structure of the software,
illustrated in Figure 28, is not dissimilar to that of a basic language like SIMAN or Arena.

A detailed description of the simulation is included in Appendix A.

Figure 28. Simulation structure

Te variables and data arrays, shown in Figure 6 (from left to right) are next defined, as they will be used as data input to save system parameters and simulation control logic. The second step is the initialization procedure, which includes the inclusion of the truck–dock assignment schedule required to provide with specific times in which the different operations are expected to be initiated. The other portion of the data input in the initialization procedure refers to the data from WMS. Most of this corresponds to the information illustrating the current state of the system, which needs to be updated at the start of every simulation run. Information that is mainly used as part of the decision making logic may be updated with less frequency. Details about the kind of data utilized are included in Section 4.4.
The simulation operates as a discrete-event simulation: it searches for the next event to be performed from a pool of the seventeen possible events shown in the center of Figure 28. After selecting this event, the clock is updated to the time that the event should be performed, which allows doing various events in the same clock time. The process continues by executing the logic of that specific event and collecting the necessary statistics. An example of an event is the “Resource Availability” which happens when system-guided resources (forklift trucks or pallet jacks) become available. This event’s logic mainly consists of performing the task selection using the rules of the TRM algorithm.

Messages from the simulation include: a check of the truck–dock assignment schedule viability as well as providing some alerts about any problems that were encountered, i.e., not enough material for replenishment. Statistics used in traditional simulation modeling, such as those related to capacity estimation and resource utilization, may also be collected over a simulation run. The simulation continues the search/perform routine for an event until the “end simulation” event is selected. The end simulation event terminates the simulation and provides summary statistics. The simulation may be re-run after making operational adjustments (e.g., number of resources, priority changes in the TRM algorithm, etcetera) in order to comply with the input schedule, being discussed in more detail at Section 4.5.

The actual simulation program consists of eighty VB sub-routines; with the use of variable arrays is one of the major elements. The use of databases would have facilitated managing the vast amount of data to be used; however, these would require a large number of queries of information continuously required by the simulation, increasing the
simulation run time. An increase in simulation time would limit the use of this tool for real-time decision making and for daily evaluation of the system.

### 4.4 The Data

Data is the backbone of SimPC. A substantial amount of SimPC has to do with different types of data used to characterize the operations and guide the different decisions at the DC. Specifically, the data can be classified in three categories: data from WMS to be updated continuously, data from WMS with minor update requirements and non-WMS data. Sample data of each type with corresponding attributes are illustrated in Figures 29, 30 and 31, respectively.

![Sample data illustration](image)

**Figure 29. WMS data for continuous update**

Examples of data requiring continuous updating are the tasks, resources and stock information (Figure 29). The tasks are the most complex set of information to process,
due to the number of attributes with different data types in terms of programming being utilized (e.g. string, float, double, integer, etc.). This corresponds to the current tasks being recorded within the system as waiting to be performed. Tasks that are in progress are also recorded, having the resource performing it as one of the attributes. This allows for the simulation to account for these tasks as already having been created. Resources specifically refer to system guided resources, forklift trucks and pallet jacks, who login into WMS. The information includes the current location of the resource, status in terms of availability and the task being performed and space for additional comments. The stock illustrates the state of the inventory, being essential for the decision of storage and retrieval locations. This information continuously changes as tasks are being performed.

![Diagram](image)

**Figure 30. WMS data requiring minor updates**

Important information in WMS is used to characterize to DC operations, for example the positions information, materials information and zone capacities for the locations in
terms of the resources (Figure 30). This data may require minor updates, which are necessary when any adjustments impacting this information are made. Positions and material information are continuously used in conjunction with the stock information for the Put-away and Picking Algorithms, representing a substantial amount of the data needed in these decisions. Meanwhile, the zone capacities are related to the TRM Algorithm, by specifying the amount of resources allowed in a specific location at any moment in time.

Figure 31. Non-WMS data

Unfortunately, not all the data are kept in WMS (Figure 31). This requires the manual incorporation other information that cannot be imported from WMS. Examples on this include: knowing the capacity that a drive-in racks has based on the pallet size utilized, knowing the material storage strategy and the number of manual resources in the different processes.
4.5 Constraint-Based DC Management

The DC operations are constrained by the number of available resources in terms of space and personnel. A truck-dock assignment schedule plan is used to organize the operations in a more predictive manner; however, there are many uncertainty elements that cannot guarantee the viability of a specific plan. Aligned to this idea, SimPC as developed in this thesis allows for an iterative process in the search for the arrangement of resources and parameters that will guarantee the schedule viability.

Specifically, the truck–dock assignment schedule plan provides information about inbound and outbound trucks. For inbound trucks, the schedule provides the time in which the truck will be at a dock to start the unloading operations, while for outbound this has to be divided in two different elements. The first element corresponds to the time the load accumulation tasks should be started, which involves the creation of tasks into WMS for the picking of the material from the racks and temporary storing them at a dock cage or buffer area. The second time element refers to the start of dock operations, which consist of loading the material into the truck. Dock operations can only be performed when the truck is at the dock.

Figure 32 illustrates the case of obtaining a non-viable truck-dock assignment plan from SimPC. A legend to understand the figure is included in Table 8. The diagram includes two schedules for the next twelve hours from the current clock time. The schedule at the top of Figure 32 represents the plan being used as input data for SimPC, while the schedule at the bottom is the result after emulation the decision making at the DC with SimPC.
Table 8. Legend for truck-dock schedule comparison

Figure 32. Truck-dock schedule comparison
In this example, the time identified as “14” represents the current clock time in which the evaluation is performed where data is gathered from WMS to obtain the current state of the DC. Each row represents the schedule of a dock, while each column delineates a different time of the day. Within each row of the schedule, the tasks required for inbound and outbound trucks are divided in two major classifications: storage/accumulation (for inbound/outbound) tasks and operations at the dock. Simulation results provide a more detailed schedule identifying the dock operations as idle, in the verification process or in charge of the loading/unloading team (LT/UT).

Arrows indicate different scenarios of the operations for this example, which include possible delays or work performed in advance. For instance, the arrow numbered “1” illustrates two reasons for delay, an extension in the accumulation tasks for outbound truck B and a delay in a previous truck (A) assigned to the same dock. The second arrow presents the opposite case, in which work is done in advance for inbound truck K, having reduced the overall operations for this truck by two time units. The arrow numbered “3” demonstrates the benefit of measuring specific tasks in the dock operations. Here it is evident that the reason for a delay are two idle times, having the first one prior to the verification process and the second one before initiating the truck loading. This means that the resources following the idle time were not available at the needed times to make this portion of the schedule viable.

Further, these details provide insight to identify the reason for non-viable schedules, which as illustrated may include having system congestion as an effect of large delays in the accumulation tasks. These delays could also be a result of having a problem at the
dock operations. The reasons for each delay may be identified by analyzing the allocation of specific idle times.

After identifying the reasons for delays, the problem shifts to determining the decisions needed to achieve the intended schedule. SimPC is focused at the tactical and operational level of the decision making process. While tactical decisions may be concerned with aspects as the dimensions of the resources and the layout arrangement requiring some investment, the operational decision are limited to assignment and control of the available resources (Rouwenhorst et al. 2000). Operational decisions should be considered first to ensure fulfillment of day to day tasks at the DC. In terms of the typical DC, this includes modifying resource assignments and altering the TRM Algorithm parameters.

In terms of the TRM Algorithm, the number of alternatives to evaluate is combinatorial, for which certain guidelines can be developed. Operational decisions will only be useful for the improvement of the accumulation procedures through modification of the TRM Algorithm since in the presence of a fixed schedule dock operations can only be improved by increasing the number of resources. Moving to the upper level of the decision making process, both the accumulation process and the operations at docks can be improved by performing better tactical decisions in which variations in the number of resources is allowed.

Measurements of the deviation from the original truck–dock assignment schedule given as part of the input data, and the schedule achieved by the simulation model, should be used for this analysis. Also, performance measures regarding the utilization of resources can contribute in the identification of solutions for these efforts.
4.6 Conclusion

This chapter demonstrates the SimPC approach, from the actual model development to the benefits that can be obtained. It was defined how SimPC is not limited to traditional simulation analysis, but could be used as part of the DC control schema. The different DC elements incorporated into SimPC were found to include: the data input, different controls and mechanisms, which in general summarize everything considered in SimPC.

Particularly, the data input has been identified as a fundamental element, being designed to obtain data in real-time. This further serves to account for the current state of the DC and have the information needed to predict its performance. This is specifically what makes the difference between traditional discrete-event simulation modeling and SimPC, by not having historical data as the base for the analysis.

In summary, SimPC was designed to emulate the DC operations through following the same rules and maintaining the same data structure.
Chapter 5

Analysis for SimPC

5.1 Introduction

This chapter provides an overview and discussions about the necessary analyses for the implementation of SimPC. This includes the validation of SimPC, understanding the correlation of the decision algorithms and defining the experiments to be performed. As part of the validation of the model, assuring the correct characterization of the resources is essential. For example, an allowance calculation of the forklift trucks to take into account breaks and personal needs is required. A method is developed that can be used to verify the correct characterization of all this resource.

In addition, having robust decision algorithms brings up the issue of how sensitive are this decisions to minor changes. For this reason, an experiment is conducted in terms of the inbound and outbound docks assignments, which affects the rules being followed by the algorithms. Finally, specific DC controllable variables are discussed in conjunction with various performance measures that should be used for the benchmarking of the DC operations.
5.2 Validation

While verification is concerned with building the model right, validation refers to building the right model in terms of its accurate representation of the real system (Hughes, 2003). To perform the validation it is important to have a good representation of the DC performance, which will then be used as a comparison in the validation procedure to calibrate the model.

However, validating all of SimPC model features simultaneously could result in a very complex task, by having highly interacting features that seem unattainable to control. Further, an exhaustive amount of data would be necessary to assure a complete sample of the real system to completely characterize the DC operations for all time periods and dynamics. This motivated simplifying the validation procedure to focus on specific SimPC elements per separate, performing multiple comparisons as suggested by Figure 33.

![Figure 33. Validation procedure](image-url)
A systematic approach is proposed in order to isolate and validate the different SimPC model components independently. This results in an iterative validation process, as illustrated by Figure 33, in which multiple adjustments allow the correct characterization of the DC’s behavior, being done systematically to the variety of simulation modules. The revisions build on top of each other, increasing the validity of the complete model, by adjusting the discrepancies between SimPC and the real DC operations.

A substantial part of the validation procedure is the validation of specific personnel resources to assure their proper characterization, corresponding to a big portion of the SimPC components. This is particularly necessary as the result of dealing with a non-automated DC, where resources are not as predictable. In automated DCs, parameters such as distance and velocity are sufficient to demonstrate the performance of the many resources. However, in this case there is the human factor, having variability from person to person. The performance depends on their knowledge, learning velocity and physical conditions, while having other personal needs that may interrupt the operations.

As illustrated in Figure 34, there are two types of resources: manual and the system guided resources. Time studies should be performed to characterize the different processes done by manual resources, obtaining performance by cases or pallets, depending on the process being analyzed. Ideally, these performance results should be in the form of a probability distribution, to better account for the expected variability. After this, the validation is accomplished through the use of allowance adjustments that serve to calibrate manual resources in the model, accounting in this way for other variation
These allowances should be calculated based on samples of complete processes in the real DC (e.g. complete unloading for a specific truck).

![Personnel Resources Diagram](image)

**Figure 34. Personnel resources**

For system guided resources, direct estimates can be calculated as these are characterized only in terms of distance and velocity. As WMS maintains historical productivity data of system-guided resources, data can be easily gathered to analyze difference among resources, for potential use for adjusting the simulation parameters. For example, in the case of forklift truck drivers, WMS provides historical data for operators through different periods of the month and year. This data was utilized for these purposes. This way, the human factor has been incorporated to forklift trucks, by having an allowance to consider the difference in operators’ capabilities.

As illustrated by Table 9, twelve data sets were tested to analyze fork truck performance. This required running the simulation such that the fork truck resource performs the tasks in the same order that was performed at the DC. An allowance value
per data set was obtained through the comparison of these simulation results with what happened against the real operations.

Table 9. Data sets for allowance calculation

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Resource</th>
<th>Creation Time</th>
<th>Number of tasks</th>
<th>Total time (hours)</th>
<th>Throughput (tasks/hour)</th>
<th>Allowance needed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Sim</td>
<td>Real</td>
<td>Diff</td>
</tr>
<tr>
<td>1</td>
<td>L18120</td>
<td>10/1/08 2:03 PM</td>
<td>107</td>
<td>4.763</td>
<td>7.170</td>
<td>2.407</td>
</tr>
<tr>
<td>2</td>
<td>L18113</td>
<td>10/1/08 6:07 AM</td>
<td>95</td>
<td>4.277</td>
<td>6.010</td>
<td>1.733</td>
</tr>
<tr>
<td>3</td>
<td>L18117</td>
<td>10/1/08 2:02 PM</td>
<td>71</td>
<td>3.206</td>
<td>4.108</td>
<td>0.974</td>
</tr>
<tr>
<td>5</td>
<td>C16872</td>
<td>4/30/08 6:08 AM</td>
<td>51</td>
<td>1.859</td>
<td>4.700</td>
<td>2.841</td>
</tr>
<tr>
<td>6</td>
<td>C16901</td>
<td>4/29/08 10:04 PM</td>
<td>68</td>
<td>2.169</td>
<td>4.866</td>
<td>2.697</td>
</tr>
<tr>
<td>7</td>
<td>L18113</td>
<td>10/15/08 6:42 AM</td>
<td>78</td>
<td>3.509</td>
<td>4.830</td>
<td>1.321</td>
</tr>
<tr>
<td>8</td>
<td>L18114</td>
<td>10/16/08 6:32 AM</td>
<td>104</td>
<td>4.509</td>
<td>6.266</td>
<td>1.757</td>
</tr>
<tr>
<td>9</td>
<td>L18171</td>
<td>10/16/08 8:40 AM</td>
<td>57</td>
<td>1.929</td>
<td>2.670</td>
<td>0.741</td>
</tr>
<tr>
<td>10</td>
<td>L18130</td>
<td>7/30/08 9:59 PM</td>
<td>83</td>
<td>2.453</td>
<td>3.850</td>
<td>1.397</td>
</tr>
<tr>
<td>12</td>
<td>L18120</td>
<td>7/31/08 2:04 PM</td>
<td>161</td>
<td>5.207</td>
<td>6.633</td>
<td>1.426</td>
</tr>
</tbody>
</table>

Max Allowance 60%
Min Allowance 22%
Average Allowance 34%

An average allowance of 34% seems to be a reasonable number since the simulation results do not include any other interruptions that may be present in daily operation, e.g., restacking of pallets, bathroom breaks, etc. In the allowance column two row values are too high (corresponding to data sets 5 and 6), and they are considered as outliers. This fact was verified with the DC personnel and these were resource on training during that time period, which justifies eliminating this values from consideration. From these results a Normal probabilistic distribution was found for the allowances with a mean of 28.9% and standard deviation of 4.38%, while not considering the values identified as outliers. A Normal Probability Plot and Histogram are included in Figures 35 and 36.
Figure 35. Normal probability plot for the forklift allowances

Figure 36. Histogram for the forklift allowances
Further analysis was performed to evaluate the confidence obtained with the used sample size of 10 allowances. The results from a Power and Sample Size for 1-Sample t test from Minitab software are summarized in Table 10. The results seem reasonable, having 89.31% confidence that a difference of 5% is detected. In addition to the information presented, it was found that to detect a 1% difference in the allowance with 95% confidentiality 252 samples are needed, which is not necessary for the purposes of this study.

Table 10. Minitab’s Power and Sample Size for 1-Sample t test for the forklift allowances

<table>
<thead>
<tr>
<th>Sample size</th>
<th>Difference in the allowance mean value evaluated</th>
<th>Confidentiality for detection</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10%</td>
<td>99.99%</td>
</tr>
<tr>
<td>10</td>
<td>5%</td>
<td>89.31%</td>
</tr>
<tr>
<td>10</td>
<td>1%</td>
<td>9.92%</td>
</tr>
</tbody>
</table>

5.3 Algorithm Impact

A simple experiment was performed with SimPC to illustrate the high correlation between the three decision making algorithms, known to play a significant role in the daily DC operations. In such a highly interconnected system, it is shown how there is an impact from the small change in the dock assignment for inbound and outbound trucks, while keeping the simple scenario of two trucks (one inbound and one outbound).

This analysis is performed based on time performance measures, in which accumulation time is considered for outbound trucks and the sojourn time for inbound trucks. The accumulation time represents the time required to perform all the retrieval
operations for the outbound truck, while the sojourn time for inbound refers to the completion of the storage tasks for the inbound trucks.

All the tasks created, for storage and retrieval, are performed by the same set of resources, mostly forklift trucks, which are scheduled by the TRM algorithm. Meanwhile, the other two algorithms, Put-away and Picking, provide the specific storage and retrieval locations, respectively, assisting the creation of the 86 inbound tasks and 89 outbound tasks. These location assignments may vary depending on the dock assignment, having different distance implications for the tasks under different scenarios.

The experiment consisted of two evaluations: 1) maintaining the inbound truck dock constant (dock 16) and changing the outbound truck assignment; and 2) changing the inbound truck dock and maintaining the outbound truck dock constant (dock 1). Both of these analyses, were performed by selecting docks to equally represent the different DC areas, using docks 1, 22, 23, 32, 35 and 45 for the perturbation of the outbound truck dock; and 3, 16, 20, and 45 for the perturbation of the inbound truck dock.

The results from this experiment are included in Tables 11 and 12, which clearly illustrate that in fact there is an impact from the dock assignment to the accumulation times and sojourn times for outbound and inbound trucks, respectively, which values are shown in the “time” column of the tables. From Table 11, one can note that a 239% increase was found from the minimum accumulation time of 26.5 minutes (dock 1) to the maximum value of 89.8 minutes (dock 23). Table 12 shows that the major difference found consists of 158%, from the inbound sojourn times of 52.7 minutes (dock 16) to 99.1 minutes (dock 45). Interestingly, in both cases the major difference is found for the truck dock being changed.
Table 11. Results with constant inbound dock

<table>
<thead>
<tr>
<th>Shipment type</th>
<th># tasks</th>
<th>Dock</th>
<th>Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>52.68</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>1</td>
<td>26.5</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>57.15</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>22</td>
<td>53.33</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>90.33</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>23</td>
<td>89.85</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>55.5</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>32</td>
<td>86.05</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>60.02</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>35</td>
<td>88.3</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>83.83</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>45</td>
<td>56.93</td>
</tr>
</tbody>
</table>

Table 12. Results with constant outbound dock

<table>
<thead>
<tr>
<th>Shipment type</th>
<th># tasks</th>
<th>Dock</th>
<th>Time (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>86</td>
<td>3</td>
<td>66.53</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>1</td>
<td>68.5</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>16</td>
<td>52.68</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>1</td>
<td>26.5</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>20</td>
<td>63.5</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>1</td>
<td>34.4</td>
</tr>
<tr>
<td>Inbound</td>
<td>86</td>
<td>45</td>
<td>99.13</td>
</tr>
<tr>
<td>Outbound</td>
<td>89</td>
<td>1</td>
<td>37.36</td>
</tr>
</tbody>
</table>
5.4 Experimental variables

The intent of this section is to define the experimental capabilities SimPC provides from the broad spectrum of necessary decisions for the design and daily operations of a DC. As discussed in Chapter 2, DCs decisions can be summarized in three levels: strategic, tactical and operations. SimPC is designed to emulate the daily processes followed in the DC, which allows the evaluation of both, tactical and operations decisions. Within these two levels, 8 decisions have been defined as interesting features to evaluate, resulting in 32 (including 3 priorities and 11 criteria weights for the TRM Algorithm) controllable/independent variables to be considered (Figure 37).

Figure 37. Controllable variables
From a design of experiments (DOE) point of view, a factorial design will result in the most complete experiment as it not only considers the effect of each factor, but allows evaluating the interactions among these. However, this increases the experimental space by having a larger number of alternatives to be considered. Also, the numbers of levels of independent variables and replications have an impact on the number of experimental conditions to manipulate. The number of levels is an independent decision per variable, while the replications refer to the repetitions of the whole experiment. For example, from the 32 controllable variables stated above, there will be a factorial design of $3^{32}$ with five replications per experiment would result in $9,265,100,944,259,200$ factorial points.

To reduce the experimental space, specific problems should be selected as part of the DOE. This allows keeping a portion of the variables constant for some of the current operating policies, while looking at multiple combinations of the variables of interest. Following this concept, three experiments are defined that could be used to explore different decisions at the DC. The identified experiments include: 1) evaluating materials and positions parameters; 2) considering different number of resources; and 3) assessing the performance of the TRM Algorithm based its different parameters.

In terms of the DOEs for this research, three levels per variable and five replications are recommended as a starting point. Having relatively small parameters for the DOE as a starting point allows understanding the performance and sensibility of the different parameters to specific performance measures, serving as the base for further experimentation. At the same time, for the purpose of initial feasibility discussion, this provides consistency with the previous example. Moreover, three levels for each control variable allows the experiment or to use the current policy as one level, while considering
setting parameter values in opposite directions for that variable, i.e., a lesser and greater value than the current policy. In the case of replications, the larger the number of replicates the greater confidence in the results. Statistical analysis should be performed to determine the appropriate number of replicates based on the desired confidence.

In terms of the first experiment mentioned, materials and positions are managed through a class-based classification (ABC) and the fixed assignment of materials to positions. While using a class-based classification may seem like a simple issue, the specific ABC classification and fixed assignments may be an interesting area to explore. For SimPC this will only require changing the parameters in the data input phase with no direct impact in the simulation code. Having only four variables (Figure 37) for materials and positions with two levels and five replications provides a $3^4$ design and 162 factorial points.

In terms of resources, six personnel resources are considered, including forklift trucks, pallet jacks, unloading teams, loading teams, verifiers and wrapping teams. Even though it is expected that with a greater number of these resources, the more work that can be performed, having the appropriate mix is essential. As illustrated in the state graph (Chapter 3), the overall performance of the DC depends on the appropriate coordination of the personnel resources, being these responsible of the different processes that interact with one another. For SimPC this does not represent major changes, having only an impact on the input data or some of the variables defining their number. A factorial design is very appropriate as it allows considering a variety combinations. Under the same experimental conditions describe above, evaluating the defined personnel resources represents a $3^6$ design with 3,645 factorial points. A more reduced
experimental design may consist of evaluating only two levels with five replicates, resulting in a $2^6$ design with 320 factorial points.

As demonstrated by the state graph analysis in Chapter 3, forklift trucks appear to be a critical operational node as they account for 11 of 23 transitions in the state graph. The TRM Algorithm assigns tasks to the forklift trucks and therefore could be the most significant component in the DC control. It operates using different parameters, with a total of 14 variables (3 priorities and 11 weights), representing variables to be changed within SimPC. These are explained in detail in Section 3.3.1. The DOE resulting for this consists of a $2^{14}$ design with 23,914,845 factorial points. A simplified version of two levels per variable will still result in a big experiment, with a $2^{14}$ design and 81,920 factorial points. Further, considering only two replicates provides 32,768 factorial points.

The size resulting for this experiment motivated looking for other simplifications opportunities, reducing the experimentation with the criteria weights to 15 scenarios (Table 13). These were found to be the criteria and sub-criteria utilized in the DC under study. The first scenario corresponds to the current operational policy, while the others are perturbations in which all the combinations for the main criteria were defined in terms of three cases: 1) only one criteria is important; 2) two of the criteria are equally important; and 3) all the three criteria are equally important. Sub-criteria were only necessary for the Static and Synchronization weight, in which the first two cases previously described were used. Finally, having these 15 scenarios combined with three levels for the three static priorities and five replicates results in 675 experimental points, which is a more reasonable analysis.
Table 13. TRM weights experiment scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>LSD Weight</th>
<th>Criteria Route Weight</th>
<th>RouteSub-criteria Weight</th>
<th>Distance Weight</th>
<th>Criteria Stat_Synch Weight</th>
<th>Static Weight</th>
<th>Synchro-Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.5</td>
<td>0</td>
<td>n/a</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.5</td>
<td>0</td>
<td>n/a</td>
<td>0.5</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.5</td>
<td>0</td>
<td>n/a</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.33</td>
<td>0.33</td>
<td>1</td>
<td>0.33</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.33</td>
<td>0.33</td>
<td>1</td>
<td>0.33</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.33</td>
<td>0.33</td>
<td>1</td>
<td>0.33</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>n/a</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>n/a</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>n/a</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>0</td>
<td>n/a</td>
<td>0</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>n/a</td>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

For the experiments described above, multiple performance measures could be obtained. Figure 38 illustrates the most relevant performance measures considered for this work, being classified in two major categories, utilization and time-related measures. For utilization, understanding of the use of resources goes from the space (e.g., docks, racks) to actual personnel (manual and system-guided). In the case of space, this helps in the understanding of potential capacity problems or inefficiencies. For resources, this could provide guidelines of the number of resources needed. Also, from a work balancing perspective this provide interesting insights.
Figure 38. Performance measures

Time related measures are very significant performance indicators to understand deviations from the planned dock schedule. Also, potential inefficiencies in terms of inbound and outbound sojourn times can be identified. Sojourn times refer to the total time an outbound or inbound truck spends waiting for the completion of all its related operations. For inbound trucks this includes from the unloading operations to the storage of the materials, while for outbound trucks goes from the start of the accumulation operations to the truck loading. This further helps verify the impact of having shared resources, especially for the system-guided resources.

5.5 Chapter Summary and Conclusions

This chapter included a discussion of some of the important analysis that should be conducted using SimPC. Validation, as for any type of simulation is an important step to be conducted. Particularly, validating that the personnel resources are aligned with the DC operations represents a major part of the validation process, having the allowance
calculation for forklift trucks as a good representation of the calibration between the SimPC and the real DC.

Through a simple experiment it was demonstrated that there is a high correlation between the different DC decision algorithms, adding special importance to the dock assignment for inbound and outbound truck. This further highlights the importance and supports the need of evaluation truck dock assignment schedules as part of the daily DC operations.

Finally, multiple experiments should be performed using the controllable variables defined in Section 5.4. The experimentation space should be narrowed down to a manageable size. A factorial design is recommended as it not only evaluates the effect of responsive variables, but look at the interaction among them. Experimentation represents learning in terms of the effect of the different DC parameters, while increasing visibility into the DC operations for better decision making.
Chapter 6

Conclusions and Further Study

6.1 Conclusions

In this thesis, a simulation modeling approach, called Simulation for Predictive Control (SimPC), was developed to characterize the operations of a large-scale and stochastic non-automated DC. SimPC, designed based on an existing DC from a Multi-national Company, incorporates a variety of control features that are necessary at any DC. Resources representing the major performance elements to be controlled were identified. These variables included the utilization of space through the use of different kinds of racks, to the task selection and assignment of the different personnel resources with detail interactions to be considered and the methods and devices that are used to store and retrieve product.

The Warehouse Control System (WCS), combined with the different decision making algorithms, provides the framework for the modeling critical decisions. Also, the WCS serve as the main source of information, showing the current DC state in real-time, which expands the scope of SimPC. SimPC is not only limited to traditional simulation analysis, being able to form part of the control of the daily DC operation, where the characterization of detail interactions of resources and decisions is essential.

In addition, this thesis presents a new approach for managing non-automated DCs with the aim of providing a tool that can account for and measure the effects of variability
There exists extensive research on the control and use of simulation for automated DCs; however, research on the development of simulation-based decision making tools for non-automated DCs is limited.

The primary focus of SimPC is to deal with the difficult issue of constraint-based DC management, where SimPC serves as a decision making tool to help evaluate the truck–dock assignment schedule viability, while facilitating the identification of the tactical and operational changes required to achieve the intended schedule.

It can be concluded, that the actual benefit of using simulation modeling is in terms of the level of detail allowed to characterize the different processes, especially when dealing with complex systems. However, this makes simulation case specific, having to potentially create a brand new simulation model for every DC, assuring the use of the same rules, data format and nomenclature for the characterization of products, resources and space in general.

6.2 Further Study

As detailed in Chapter 5, additional experimentation with SimPC is still needed. The data gathering for validation should be completed, to finish the calibration of SimPC with the DC. After this phase, the same data already gathered for validation should be use for traditional simulation analysis in the development of benchmarks when perturbing the parameters, while identifying the significant of the different controllable variables into the DC performance measures. This includes running multiple replications for the validity of the experiment.
For the implementation of SimPC as part of the DC control schema, additional implementation work should also be performed. The limitations of this tool are based on the ability to obtain the initialization information and being able to represent manual resources not being tracked by the system. A vast amount of information is needed, requiring more work from an implementation standpoint. Automated procedures with interfaces between the simulation and the system should be considered to facilitate the data gathering for the initialization process, making this a real-time procedure. For greater model accuracy, the status of manual operations should be tracked by the system. Having the start/finish times of manual operations allows calculating the remaining time from the work characteristics and time study results.

User-friendly interfaces should be developed for the use of this tool in an iterative process as is its aim in the schedule viability evaluation process. This will also require a detail analysis of the controllable variables affecting the truck-dock schedule, to be used as guidelines for the simulation users. Further, learning capabilities could be incorporated with the aim of developing rule-based algorithms to decide the system modifications needed to eliminate schedule variations.

Particularly, the concept of SimPC should be generalized to other applications, serving as a forward looking technique in the evaluation of schedule viability.
Appendix A

Pilot Simulation Description

1.1 Start Simulation

The “Start Simulation” routine is the main procedure of the simulation model, which triggers the initialization procedure as a starting point. Depending on the next event to be performed, decided by the timing sub-routine, it directs the simulation to perform it addressing the correct sub-routine to be used. It directly triggers the following sub-routines:

- Initialization
- Timing
- Resource availability
- Resource task origin coordinate update
- Resource task destination coordinate update
- Sided truck arrival
- Availability for unloading sided
- Sided truck verifier availability
- Inbound bulk-truck arrival
- Unloading team availability
- Inbound task creation
- Verifier availability
- Loading team availability
- Outbound truck arrival
- Outbound start accumulation
- Outbound task creation
- Dock availability
- Wrapping team availability
- End Simulation

1.2 Initialization

The initialization procedure provides initial values to most of the variables and data arrays. The following sub-routines are triggered as part of the initialization procedure:

- Obtain data – allows saving the text files data in a specific order, which is important for some of the logic work
- Task system duration calculation – calculates the task duration as it is assumed by the system, being information necessary to have for the TRM algorithm. Through this procedure the value is calculated only once, and not every time the algorithm is performed.
- Initialize next available resource time – the initial available time for resources is set up as the time the simulation starts.

1.2.1 Obtain data

Data input is obtained from text files into the simulation model. This is a substantial part of the initialization procedure, in which text files are being read and saved into data arrays. Most of these files represent the information was obtained from WMS, mostly keeping the same data format. Also the schedule information, which is expected to be obtained from the optimization models, is input in the same manner. The greatest challenge in this procedure corresponded to having many different data types within one file of data, which required defining everything in general as an “object” type and then defining each one independently with its corresponding type.

The order in which this information is saved is also important; since in this data saving process other information for simulation control is obtained. For example, the aisle capacity array is adjusted as the resources and tasks information is being saved in order to have the actual state of the capacity constraints. The data being considered includes the information defined in Appendix C, and other control arrays. This includes:

- capacity zones array
- tasks array
- resources array
- positions information array
- materials information array
- material volume array
- stock array
- inbound schedule array
- inbound material and quantity array
- outbound start accumulation schedule array
- outbound material and quantity array

1.3 Timing

The “Timing” sub-routine, which is triggered every time an even is finished, determines the next event type to be performed by selecting the event with smallest time value. This is done based on the values of the corresponding data arrays for each of the event types. As part of this procedure the clock is updated based on the next event to be performed with the smaller time. Initially, the simulation reads the clock time from the computer being used to run the simulation. A date and time format is being used to provide more accuracy in the time parameters and to comply with WMS data format.

1.4 Resource Availability Event

The resource availability event deals with the management of system guided resources and their decision making to perform the next tasks and subsequently schedule the updates needed when arriving at the origin and destination of the task. Most of this event deals with the selection of the next task being performed by the TRM algorithm explained in detail at section 1.6.3.

The first step of this event logic corresponds to verifying if the resource already has a task assigned (“Determine tasks assigned” sub routine). If the resource has a task
assigned, that task will be performed first. If the resource has various tasks assigned, it will be performed by the order of the task number, selecting first the task with the smallest task number. This solves the initialization problem of determining the initial status of the resource. In the case that the resource is in the middle of a task, the simulation assumes that this task is being initiated upon the start of the simulation run. This is needed due to the manual assignments that may be performed by the TRM employees. Besides that, for the continuation of the simulation no manual assignments are performed assuming that these resources are being completely guided by the system that follows the TRM algorithm as the decision logic. If at the initial stage the resource has no task assigned, then the TRM algorithm is performed as well (“TRM algorithm” sub-routine).

While performing the TRM algorithm, explained in section 1.6.3, the task feasibility plays a significant role in the whole process. For this reason, the distinction between forklift trucks and pallet jacks is crucial. The code for this algorithm can be summarized in six sub-routines, being most imbedded within the other in the following form:

- TRM algorithm
  - LSD priority time to origin calculation logic
    - LSD and synchronization priority calculation
    - Time to task origin calculation
      - Forklift truck distance calculation
      - Pallet jack distance calculation

The “TRM algorithm” sub-routine performs final priority calculation and the final task selection after all feasible tasks have been considered. The “LSD priority time to origin calculation logic” sub-routine provides the logic to differentiate between forklift trucks and pallet jacks while the “LSD and synchronization priority calculation” calculate the values of the specific priorities. The “Time to task origin calculation” sub-routine calculate the time it will take the resource to travel to the origin location of the task, which is information needed to calculate the distance priority. Depending of the resource type, the “Forklift truck distance calculation” sub-routine or “Pallet jack distance calculation” sub-routine is used.

After a task has been selected, the capacity constraints are updated through the “Update capacity task assignment” sub-routine such that the current assignment is considered. Then, the “Resource origin coordinates next” event, “Resource destination coordinates next” event and the “Resource availability” event are scheduled depending on the resource status (to be 0 or 1), which is decided in the TRM algorithm considering the available and feasible tasks, and the type of resource. These events are scheduled based on the tasks selected to be performed by the TRM Algorithm having specific distance characteristics.

There are only two types of resources: forklift trucks and pallet jacks, being both electronic and guided through screens or handhelds. A resource status of one means that at least one task could be performed by the resource. However, when the resource status is zero, it is meant that the TRM algorithm procedure was not able to find a task that this resource could perform either because there are no tasks in the systems or none of the ones available are feasible.
Feasibility of schedules and due dates may be attributed to various reasons. It may be a result of the tasks characteristics in which every resource has specific task types that is allowed to perform. For example, the resource may be a forklift truck, but all the tasks in the system may be case picking which are intended to be performed by pallet jacks. Also, there may be tasks available tasks, but there may be already assigned to other resources. Furthermore, there is also the case in which the origin or destination location of the available tasks may be constrained by the aisle capacity requirements. For this the DC locations are divided by zones, having location grouped within zones.

1.5 Resource Origin Coordinates Next Event
This event performs the updates needed when a resource has arrived to the task origin location. An important update corresponds to the aisle capacity constraints in which the data regarding the number of resources per zone is updated by specifying that the resource is no longer in its previous location through the “update capacity new location task origin” sub-routine. A differentiation is made between the tasks assigned by the system through the TRM algorithm and the ones that were assigned manually being found to be assigned in the initialized task data. For the first type of tasks, the aisle capacity is adjusted as described; however, the second type depends of an additional verification. Through the use of the “Check other tasks characteristics task origin” sub-routine the simulation verifies if the resource is expected to be return to this zone as part of other task assignment for which the capacity constraint still holds. If there is no other task assignment with this zone, then the zone capacity is updated. Also, in the first iteration of this same kind of resource task assignment, the aisle capacity update is not perform since the resource registered location was not considered in the aisle capacity initialization. Furthermore, the current location of the resource is also changed, to be the origin location coordinates of the task being performed.

1.6 Resource Destination Coordinates Next Event
The updates required when the resource arrive the destination of task being performed are very similar to the ones done for the arrival to the task’s origin location. The aisle capacity updates are performed the same way, with the exception that in the first iteration the resources with manual assignments do check if the update is needed through the Check other tasks characteristics task destination” sub-routine. For system task assignments, the “update capacity new location task destination” sub-routine is used. Also, the current location of the resource is updated to be the task destination location.

1.7 Aisle Capacity Constraints
The capacity constraints, which are mostly applied for forklift trucks, are initialized with the input of tasks and the input of resources information sub-routines. These constraints are updated when assigning a task and when the resource changes the location by arriving the task origin or the task destination. A difference has to be made between tasks that have been assigned manually by an employee into the system and the ones assigned using the TRM algorithm.

The complexity of applying these constraints was mostly due to having manually assigned system guided resources to specific tasks. This caused not knowing how the system behaves under these circumstances in which it may not considered these
assignments at all, ignoring that there will be resources in these areas or by the other hand it could be the case in which it considers each task assigned independently as having resources in all those areas while not reflecting that many of these may be assigned to the same resource.

For the simulation model, the highest intelligence was assumed, in which these manual assignments are considered, in which a resource assigned multiple times to the same zone is being counted only once. Also, if the origin and destination of the task have the same zone, this will also be considered only once. Assuring that this rule are followed becomes more important while tasks are being performed in which different scenarios may be possible. Various data arrays are kept to control what resources have been considered to what zones. In the case of tasks assigned by the system using the TRM algorithm, the logic is much straight forward, by just updating the change of these accordingly as tasks are assigned and forklift truck locations are updated.

1.8 Stock Update

The stock is being updated in terms of available material when there are inbound and outbound operations. Initially, the tasks taken from the system already had performed this stock updated. When there is an inbound operation this stock is updated to assure that no other material is added to that location, while assuring the capacity of the rack if the same material is considered. In the case of outbound operations, the selected material is eliminated from the storage location, such that it is not used for any other shipment. Moreover, the stock update is not performed in a specific sub-routine, but updated at the different events within the simulation as necessary.

1.9 Inbound Operations

There are two types of inbound operations, the ones for sided trucks and the ones for bulk trucks. The sided trucks are received continuously from the nearby plants. The difficulty encountered with this event was in terms of data available to perform accurate predictions of the future. The DC does not have any data of the trucks they will be receiving in terms of timing, materials and quantities, which substantially may affect the accuracy of the simulation. This is because depending of the material SKU and quantity, there will be different decisions in terms of Put-away Algorithm and the added congestion affecting the TRM algorithm decisions. In terms of the simulation, it was coded as if this information was available, and in the case it is not assumptions need to be made. By the other hand, the information regarding bulk inbound is available, since most of this trucks have been waiting in the parking lot for days before they are unloaded.

1.9.1 Inbound sided-truck events

Sided trucks require a gas forklift truck to perform the unloading in pallets and a verifier to assure these are correct before the tasks can be created. Three events are used to coordinate these activities:

- Inbound sided truck arrival
- Availability for unloading sided
- Sided truck verifier availability

These events interact with one another since one may perform the scheduling of the other. Upon the arrival of a sided truck it is verified if unloading resources are available
to perform this task. If they are, the availability for unloading sided event is scheduled to perform the unloading at that specific moment in time. However, if no one is available, the sided truck is added to a waiting list for this resource. The availability for unloading sided consists of the gas forklift trucks moving the pallets from the truck to the DC, for which it is given a processing time dependant of the number of pallets to be unloaded. After all the pallets from the same truck are confirmed to be loaded, which happens at the unloading resource next available time, it is verified if a sided truck verifier is available. In the case a sided truck verifier is available; the resource is scheduled for that time. If there is no availability, the request is added to a waiting list. When the verification finishes, known to be at the next verifier availability, the inbound task creation event is scheduled for the same verification ending time.

1.9.2 Inbound bulk-truck events

Bulk trucks require an unloading and palletization process, which is performed by the same kind of resources simultaneously. For this reason two main events control the process for the material received from bulk-trucks:
- Inbound bulk truck arrival
- Unloading team availability

Upon the arrival of an inbound bulk-truck, which is assumed to be at the same time given by the schedule, it is verified if its assigned dock is available to start the operations. In the case that the dock is not available, the truck is added to a waiting list. If the dock is available, it is assigned to the dock and the availability of an unloading team to be scheduled is verified. Following the same procedure, if the unloading team is available, it is triggered to start the process at that moment; however, if it is not available, the request is added to a waiting list. The unloading team time is used as a function of the number of cases and pallets being formed. When the unloading team finishes, the inbound task creation event is scheduled, not for the same ending time of the unloading and palletization process, but for a later time using an allowance because in the real operations this does not occur immediately.

1.9.3 Inbound task creation event

In this procedure, the Put-away algorithm is used as the rule to create the tasks, following the algorithm characteristics, explained in section 1.6.1. Many sub-routines are used to achieve this. First, to define the specifics of the tasks being created, the following sub-routines are used:
- Determine dock origin information – the dock number, coordinates, and zone are found.
- Determine material information – the material’s characteristics, such as the search rotation (A, B, C), storage requirements (E1, E2, etc.), storage constraints (DI), number of cases allowed per pallet, and pallet type (PL1 or PL2).
- Determine search rotations – depending on the material search rotation, the order in which the different locations will be search is determined

For complete pallets and partial pallets, different procedures are followed. Complete pallets, if allowed by the material information, are mainly stored in drive-in racks unless replenishment is needed. The procedure for complete pallets starts with the “Replenishment needs check” sub-routine. Here is it verified whether a case picking or
the first level of a case picking rack is empty and there are not tasks already schedule for this, meaning that there was no material available in the warehouse to perform this. In the case there is such empty locations, these tasks are created. As described by the DC personnel, this is rare to happen, however, this was informed as the logic being followed by the algorithm.

If there is no replenishment need, and if the material entrance strategy is E1 and drive-in storage is allowed, different drive-in storage areas are verified following a specific search order. This rules are given at “In storage area search order” sub-routine. As a starting point, the preferred storage area is calculated through the “Balance areas calculation” sub-routine (including within it the “Calculate Number of pallets per area” sub-routine), which determines the percentage of each material at the three storage areas, known to be 1, 2 and 3. After this, the procedure changes the search rotations accordingly based on the order it should search the storage areas and the rotations, which follows the rules described in section 1.6.1. Moreover, this includes having an initial preference on partially full locations. The actual search for the location is given by the “Drive-in inbound search rules” sub-routine, which is triggered by the “In storage area search order” sub-routine every time a change is made in the search rules, until an assignment has been achieved.

The “Drive-in inbound search rules” sub-routine looks at every drive-in storage location independently in the order given by the initial data order. If the location complies with the characteristics previously defined in the “In storage area search order” sub-routine, which also includes the feasibility of storing the material at this location, the pallet space is calculated through the “Drive-in Inbound” sub-routine. Here, if there is space at that location, the simulation proceeds to create these tasks through the “In task creation” sub-routine, for as many pallets as possible that could be accommodated in the same location. If there is no space at all, the next location to evaluate is given by the “Drive-in inbound search rules” sub-routine. The “In task creation” sub-routine adds a row into the tasks variable array, including the twenty-seven items of information that this requires.

If the material is not E1, then it follows the “Not consumo material” sub-routine is followed. Materials characterized with a feasible drive-in storage, are storage in this locations if possible, using the “In task creation” sub-routine. By the other hand, if drive-in storage is not allowed, the single deep racks storage location, levels 3-6, are searched creating the tasks with the “In task creation” sub-routine.

Partial pallets are always stored in single-deep racks, location being found by the “Partial pallets in 433” sub-routine. Here, the single deep rack locations, levels 3-6, are the only ones considered. The feasible locations will be the ones being empty and having the same material entrance strategy. For example, if the material entrance strategy is E1, it can only be stored in areas 1, 2 and 3. The tasks are created using the “In task creation” sub-routine.

Finally, the task system duration is calculated, being information needed for the TRM algorithm to be performed in the resource availability event.

1.10 Outbound Operations

The peculiarity of the outbound operations is that the truck does not necessarily need to be at the DC to start the accumulation operations. For this reason, it is necessary to
keep track of truck arrival times and accumulation start times as given in the schedule from the optimization models. This result in two types of events: the outbound truck arrival and the outbound start accumulation.

The outbound truck arrival assumes that the truck arrives at the given time by the schedule. Upon the arrival of the truck it is verified if its assigned dock is available. In the case it is, the status of its materials is verified, which mainly depends on the time the accumulation was scheduled to start, which may be the same time the truck arrival was schedule or before this time. This may require the truck to wait to be loaded for the completion of the accumulation process, a portion of it, or nothing at all. So, this event basically makes the truck available to be loaded whenever it can be moved into its assigned dock.

The outbound start accumulation event provides the time in which the task should be created. However, verification should be performed to assure that the area assigned for this accumulation is empty. In the case it is not empty; the start for accumulation is delayed waiting until the location becomes available. When the area is available, the event to create tasks is scheduled.

After the accumulation procedure has been finished, the load is ready to be verified. This is verified every time a task is being performed, consisting of checking if this is the last task for this shipment. To be verified, the verifier resource is needed, being schedule if it’s available or part of a waiting list if not. The required time for verification being used mainly depends on the load characteristics, in which the ones that have been wrapped are a function of the number of pallets, while the others are a function of the number of cases. When the verification procedure is finished, a loading team is needed. If this team is available, it is schedule for that specific clock time. If there is no loading team available, then the load is added to a waiting list. The loading time has the same characteristics of the verification time, in which it is a function of the number of pallets or cases depending of the load type.

1.10.1 Outbound task creation

The outbound task creation mainly consists of following the Picking Algorithm as described in section 1.6.2. Initialization information is obtained from various sub-routines:

- Determine accumulation destination information – find the dock information for which the shipment will be placed and its coordinates characteristics
- Determine material information – the same sub-routine used for the inbound task creation to find the details about the material being evaluated.

Initially storage are 410 is considered since this corresponds to the area designed to have material ready for easy retrieval, however, this area is not being used in the current operations. The storage area 433 is evaluated, since there is a preference to empty the single deep racks from levels 3-6 if it is possible. If the material is not found in any of these locations, the search is performed differently if the amount of material needed is more than a complete pallet or less than a complete pallet.

In the case of having a number of cases requested that is greater than the number of cases per pallet, the “Complete Pallet Retrieval Search Rules” sub-routine is followed. In this procedure, the storage area 420 is search first, through the “Search 420” sub-routine, if the material is characteristics whoe that it is allowed to be stored in drive-in
racks. If the material was not found at a drive-in rack or its storage here is not feasible, the second level of the single deep racks (storage areas 430, 431 and 432) are searched through the following sub-routines: “Search 430”, “Search 431” and “Search 432”. However, a different search order may be followed depending on the outbound dock assignment. These search sub-routines (for 430, 431 and 433) are very similar, since all look for this material in all the locations and if the material is found, the “Out accumulation task creation” sub-routine is triggered to have the task added into the task variable array.

The “Search 420” is somehow different, since for drive-in racks, the FEFO condition is followed. This required calculating the window of tasks’s expiration dates to be considered and using the “Determine possible retrieval locations” sub-routine to specific which storage location comply with the calculated FEFO window. If possible retrieval location were found, the “Determine minimum distance retrieval in FEFO window” sub-routine is used to find the location the nearest location to the dock while complying with the time window constraint. After this, the “Out accumulation task creation” sub-routine is performed. Then, through the “Determine next oldest pallet date” sub-routine, the control variable within the position file is updated.

When the requests becomes to be less than a complete pallet, this materials will be obtain as part of a case picking operation, requiring to determine the material volume, through a sub-routine with this same name, to form the pallets appropriately. The “case picking retrieval search rules” sub-routine is followed to determine the location in which this materials will be gathered. Depending on the dock assignment for the shipment, a different order to evaluate the storage locations (330, 331 and 332) is followed. The sub-routines for this are: “Search 330”, “Search 331” and “Search 332”. These procedures avoid assigning case picking locations with complete pallets in order to empty the different locations as soon as possible, for replenishment efficiency purposes. Also, the logic tries to obtain all the materials required with the same SKU from the same location if it is possible. A variable array is saved with these location assignment to be used in the formation of case picking pallets.

After all the materials have been considered, resulting in a potential list of materials needing case picking, the procedure to form the case picking pallets is performed through the “case picking tasks” sub-routine. Here, case picking pallets are formed following a specific order by aisles and assuring that volume constraints are followed. If all the cases of a specific material do not fit, these are left for the next pallet formation and the evaluated pallet considered to be finished. In this same procedure, as pallets are formed, the tasks are created.

After all outbound tasks have been created; the “task system duration calculation” sub-routine is performed, as information needed for the TRM algorithm selection of tasks and the performing of the tasks.

1.11 Dock availability

The dock availability event is schedule when a dock becomes empty after an outbound or inbound truck has finished its operations. The purpose of this event is to verify if there are any trucks waiting for this specific dock and schedule its other event accordingly to start the operations for that truck. This may be either for inbound or outbound trucks.
1.12 Wrapping team availability

When materials are moved to the buffer areas, which position coding starts with a “B” letter, a wrapping procedure is required. Based on the interviews performed to the DC personnel, it was understood that all the materials moved to this buffer areas required a wrapping process. Wrapping teams are the ones performing this wrapping procedure. In order to move materials to these areas it is necessary to have the accumulation schedule indicating one of the buffer locations. Upon the movement of a pallet to the buffer area, it is verified whether a wrapping team in available. If a resource is available, the wrapping team is schedule to perform this. In the case, not teams are available; the request is added to a waiting list. The wrapping team work consists of the time it takes to move the pallet to the machine, performing the wrapping and unloading it from the machine.
Sub Start_Simulation()

Dim Simulation_Time_Seconds As Double
Dim Simulation_Time_Minutes As Double
Dim j As Integer

Initialization()

Do
    Timing()
    Select Case NextEventType_Index
        Case 1
            ResourceAvailability()
        Case 2
            ResourceTaskOrigin_CoordUpdate()
        Case 3
            ResourceTaskDestination_CoordUpdate()
        Case 4
            OutboundTaskCreation()
        Case 5
            For j = 0 To MaxTasksIndex
                Simulation_Time_Seconds = DateDiff(DateInterval.Second, Date_Time_Now, DateTime.Now)
                Simulation_Time_Minutes = DateDiff(DateInterval.Minute, Date_Time_Now, DateTime.Now)
                Stop
            Next j
        Case 6
            InboundSidedTruckArrival()
        Case 7
            UnloadingGasForkliftsAvailability()
        Case 8
            SidedTruckVerifierAvailability()
    End Select
Loop
Case 9
    InboundBulkTruckArrival()

Case 10
    UnloadingTeamAvailability()

Case 11
    OutboundVerifierAvailability()

Case 12
    LoadingTeamAvailability()

Case 13
    OutboundTruckArrival()

Case 14
    OutboundStartAccumulation()

Case 15
    DockAvailability()

Case 16
    WrappingTeamAvailability()

End Select
Loop
End Sub
Appendix C

Initialization Sub-routine

Sub Initialization()

Vel_Forklift = 2.4 'm/s
Vel_PalletJack = 2.4 'm/s
VerticalVel_Forklift = 0.4 'm/s
DriveInEntranceVel_Forklift = 0.4 'm/s

maxtaskDurationCalc = 0

tnow = Date_Time_Now

NumOfTasksPermitted = 20000
numEvents = 16
Obtain_Data()

MaxUnloadingTeamsResources = 1
MaxSidedTruckVerifiersResources = 1
maxUnloadingGasForklifts = 0
MaxOutboundTruckVerifiersResources = 10
MaxLoadingTeamsResources = 20

All_Status_Initialize()

TaskSystemDurationCALC()

DockStatus(16, 0) = 0
DockStatus(29, 0) = 0

UnloadingGasForkliftsStatus(0, 0) = 0
UnloadingGasForkliftsStatus(0, 1) = 16

timeNextEvent(0) = Date_Time_Now.AddHours(1444)
Initiate_NextAvailable_Resource_Time()
timeNextEvent(4) = Date_Time_Now.AddHours(144)
timeNextEvent(5) = Date_Time_Now.AddHours(10)
timeNextEvent(4) = Date_Time_Now.AddHours(1444)
timeNextEvent(15) = Date_Time_Now.AddHours(1444)
ForkliftPositionTime = 10
PalletJackPositionTime = 10

CTW_HourIn_Inbound = 1800
CTW_HourIn_Outbound = 1800
CTW_HourIn_Replenishment = 1800
CTW_HourIn_Movement = 1800
CTW_HourIn_TransferReg = 1800
CTW_HourIn_AdditionalLogServices = 1800

CTW_EndHour_Inbound = 1
CTW_EndHour_Outbound = 1
CTW_EndHour_Replenishment = 14400
CTW_EndHour_Movement = 1
CTW_EndHour_TransferReg = 1
CTW_EndHour_AdditionalLogServices = 1

Static_Inbound_priority = 30
Static_Outbound_priority = 30
Static_Replenishment_priority = 60
Static_Movement_priority = 0
Static_TransferReg_priority = 0
Static_AdditionalLogServices_priority = 0

LSD_weight = 0
Route_weight = 0.5
Distance_weight = 1

Stat_Synch_weight = 0.5
Static_weight = 1
Synchro_weight = 0

StartTaskDuration_Index = 0

counterTaskNewInfo = 0

FEFOdays = 1

CasePickingMaxVolume = 1.44

XcoordPALVZY = 283.3
YcoordPALVZY = 30.426
XcoordPALVZX = 109.3
YcoordPALVZX = 30.426
XcoordPALVZZ = 459.692
YcoordPALVZZ = 1

End Sub
Appendix D

Timing Sub-routine

Sub Timing()

    Dim selectedNExtEvent As Integer = 0
    Dim r As Integer
    Dim minNextEventTime As DateTime

    minNextEventTime = Date_Time_Now.AddHours(1444)
    timeNextEvent(1) = Date_Time_Now.AddHours(140)
    timeNextEvent(2) = Date_Time_Now.AddHours(140)
    timeNextEvent(3) = Date_Time_Now.AddHours(140)
    timeNextEvent(6) = Date_Time_Now.AddHours(140)
    timeNextEvent(7) = Date_Time_Now.AddHours(140)
    timeNextEvent(8) = Date_Time_Now.AddHours(140)
    timeNextEvent(9) = Date_Time_Now.AddHours(140)
    timeNextEvent(10) = Date_Time_Now.AddHours(140)
    timeNextEvent(11) = Date_Time_Now.AddHours(140)
    timeNextEvent(12) = Date_Time_Now.AddHours(140)
    timeNextEvent(13) = Date_Time_Now.AddHours(140)
    timeNextEvent(14) = Date_Time_Now.AddHours(140)
    timeNextEvent(16) = Date_Time_Now.AddHours(140)

    NextEventType_Index = 0

    For r = 0 To MaxResourcesIndex
        If ResAvailabilityNextEvent(r) < timeNextEvent(1) Then
            timeNextEvent(1) = ResAvailabilityNextEvent(r)
            NextResAvailable_Index = r
        End If
    Next r

    For r = 0 To MaxResourcesIndex
        If ResOriginCoordsNextEvent(r) < timeNextEvent(2) Then
            timeNextEvent(2) = ResOriginCoordsNextEvent(r)
            NextResOriginCoords_Index = r
        End If
    Next r

    For r = 0 To MaxResourcesIndex
        If ResDestCoordsNextEvent(r) < timeNextEvent(3) Then
            timeNextEvent(3) = ResDestCoordsNextEvent(r)
            NextResDestCoords_Index = r
        End If
    Next r
For r = 0 To MaxIN_Sided_ScheduleIndex
  If IN_Sided_Schedule(r, 2) < timeNextEvent(6) Then
    timeNextEvent(6) = IN_Sided_Schedule(r, 2)
    NextIN_Sided_Truck_Index = r
    NextIN_Sided_Truck_Number = IN_Sided_Schedule(r, 0)
  End If
Next r

For r = 0 To maxUnloadingGasForklifts
  If UnloadingGasForkliftsNextEvent(r) < timeNextEvent(7) Then
    timeNextEvent(7) = UnloadingGasForkliftsNextEvent(r)
    NextIN_UnloadingGasForklift_Truck_Number = r
  End If
Next r

For r = 0 To MaxSidedTruckVerifiersResources
  If SidedTruckVerifierNextEvent(r) < timeNextEvent(8) Then
    timeNextEvent(8) = SidedTruckVerifierNextEvent(r)
    NextSidedTruckVerifier_Index = r
  End If
Next r

For r = 0 To MaxIN_Bulk_ScheduleIndex
  If IN_Bulk_Schedule(r, 2) < timeNextEvent(9) Then
    timeNextEvent(9) = IN_Bulk_Schedule(r, 2)
    NextIN_Bulk_Truck_Index = r
    NextIN_Bulk_Truck_Number = IN_Bulk_Schedule(r, 0)
  End If
Next r

For r = 0 To MaxUnloadingTeamsResources
  If UnloadingTeamNextEvent(r) < timeNextEvent(10) Then
    timeNextEvent(10) = UnloadingTeamNextEvent(r)
    NextUnloadingTeam_Index = r
  End If
Next r

For r = 0 To MaxOutboundTruckVerifiersResources
  If OutboundTruckVerifierNextEvent(r) < timeNextEvent(11) Then
    timeNextEvent(11) = OutboundTruckVerifierNextEvent(r)
    NextOutboundTruckVerifier_Index = r
  End If
Next r

For r = 0 To MaxLoadingTeamsResources
  If LoadingTeamNextEvent(r) < timeNextEvent(12) Then
    timeNextEvent(12) = LoadingTeamNextEvent(r)
    NextLoadingTeam_Index = r
  End If
Next r
For r = 0 To MaxOUT_Truck_ScheduleIndex
    If OUT_Truck_Schedule(r, 2) < timeNextEvent(13) Then
        timeNextEvent(13) = OUT_Truck_Schedule(r, 2)
        NextOUT_TruckStart_Index = r
        NextOUT_TruckStart_Number = OUT_Truck_Schedule(r, 0)
    End If
Next r

For r = 0 To MaxOUT_Accum_ScheduleIndex
    If OUT_Accum_Schedule(r, 2) < timeNextEvent(14) Then
        timeNextEvent(14) = OUT_Accum_Schedule(r, 2)
        NextOUT_AccumStart_Index = r
        NextOUT_AccumStart_Number = OUT_Accum_Schedule(r, 0)
    End If
Next r

For r = 0 To numEvents
    If timeNextEvent(r) < minNextEventTime Then
        minNextEventTime = timeNextEvent(r)
        NextEventType_Index = r
        selectedNExtEvent = 1
    End If
Next r

If selectedNExtEvent = 0 Then
End If

If NextEventType_Index = 0 Then
End If

tnow = minNextEventTime

End Sub
Bibliography


Dunkin, A. E. 1989, Analysis and design of storage and retrieval systems for tote size loads, Technical Report MHRC-TD-88-01, Georgia Institute of Technology, Atlanta, GA.


Hughes, H. D., 2003, Verification and Validation, CSE 808 Modeling and Discrete Simulation, Michigan State University.


SAP Library Release 6.0, 2007, SAP NetWeaver. SAP AG.


