Waveless Picking: Managing the System and Making the Case for Adoption and Change

By
G. Todd Bishop
B.S., Electrical Engineering
Brigham Young University, 2006

Submitted to the Sloan School of Management and the
Engineering Systems Division
In Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Engineering Systems
and
Master of Business Administration

In Conjunction with the Leaders for Global Operations (LGO) Program at the

Massachusetts Institute of Technology
June 2010

Signature of Author____
Sloan School of Management, and Engineering Systems Division
May 7, 2010

Certified by____
Dr. Jérémie Gallien, Thesis Supervisor
Associate Professor of Operations Management, MIT Sloan School of Management

Certified by____
Dr. David Simchi-Levi, Thesis Supervisor
Professor, Engineering Systems Division and Civil & Environmental Engineering

Accepted by____
Debbie Berechman
Executive Director of MBA Program, MIT Sloan School of Management

Accepted by____
Nancy Leveson
Chair, Engineering Systems Division Education Committee

[This page intentionally left blank]

Waveless Picking: Managing the System and Making the Case for Adoption and Change

By G. Todd Bishop

Submitted to the MIT Sloan School of Management and the Engineering Systems Division
on May 7, 2010 in partial fulfillment of the requirements for the degrees of
Master of Business Administration and Master of Science in Engineering Systems

Abstract

Wave-based picking systems have been used as the standard for warehouse order fulfillment for many years. Waveless picking has emerged in recent years as an alternative pick scheduling system, with proponents touting the productivity and throughput gains within such a system. This paper analyzes in more depth the differences between these two types of systems, and offers insight into the comparative advantages and disadvantages of each. While a select few pieces of literature perform some analyses of wave vs. waveless picking, this paper uses a case-study of a waveless picking system in an Amazon.com fulfillment center as a model for how to manage a waveless system once it has been adopted. Optimization methods for decreasing chute-dwell time and increasing throughput by utilizing tote prioritization are also performed using discrete-simulation modeling. The analysis concludes that managing waveless picking warehouse flow by controlling the allowable quantity of partially picked orders to match downstream chute capacity can lead to reduced control over cycle times and customer experience. Suggestions are also made on possible future research for how to optimally implement a cycle-time controlled system.

This thesis is the result of work done during a 6.5 month LGO internship at an Amazon.com Fulfillment Center in Lexington, Kentucky.

Thesis Supervisor: Jérémie Gallien
Title: Associate Professor of Operations Management

Thesis Supervisor: David Simchi-Levi
Title: Professor, Engineering Systems Division and Civil & Environmental Engineering

[This page intentionally left blank]

Acknowledgements

I would like to thank Amazon.com and the LEX1 fulfillment center for welcoming me into their organization and giving me the opportunity to participate in an engaging and meaningful project.

In particular, I would like to thank the following people without whose help I could not have completed the internship project:

- Akash Chauhan for being my mentor and providing direction, advice, and constructive feedback for the betterment of both my internship project and my personal career.
- Brooke Kahl for the continuous help in pushing forward initiatives that otherwise may not have been completed, and for assisting in constructing persuasive arguments for controversial pieces of the project.
- Rudy Darmawan for supporting me with all the data I could ever want, and helping on short-term notice with organizing necessary experiments for project success.
- Cherie Wong for her help in getting up to speed with system functionality and for her open mind regarding project outcomes.
- Paul Kreger and the other operations managers, who allowed me to significantly impact their pick productivity numbers as various system settings were tested, allowing me to reach the conclusions made in this thesis.
- The facilities team, who helped complete some process improvement efforts, and who were always willing to lend me their expertise knowledge
- The AFE area managers, Joel Brown, Landon Hutchison, Jeff Rieke, and others, who let me experiment in their area and on occasion assisted with testing despite their other responsibilities and busy schedules.
- Mike Passales for bringing me on to this project and for offering me the chance to participate in kaizen teams during Shingijustu kaizen week at the Lexington warehouse.
- The Leaders for Global Operations program for its support of this work

I would also like to thank Professor Jérémie Gallien, for providing insights and advice from his past experiences, helping to move the project in the right direction much faster and more efficiently than I otherwise would have been able to do, and Professor Simchi-Levi for the internship preparation that he gave.

Finally, I would like to thank my wife and kids for supporting me through the long hours and unexpected developments throughout the internship and my time at MIT. I appreciate the sacrifice that four moves within two years on a family of four has been, along with living in close quarters so that I could have this experience as an LGO student. I am fortunate to have brought them along for the ride, and could not have been as dedicated to the internship and classes without having their support.

[This page intentionally left blank]

Table of Contents

[This page intentionally left blank]

List of Figures

List of Tables

List of Equations

[This page intentionally left blank]

Chapter 1: Introduction and Overview

The growth of the global economy and global interconnectivity has increased dramatically over the past decade, with new markets opening each year. While global reach has expanded accessible markets and allowed companies to benefit from the increased demand, it has also leveled the playing field across companies by acting as the great equalizer, and has thus given power to the consumers (Friedman 2005). The significance of this is that consumers now have pricing and availability information at their fingertips, and this has put enormous cost pressure on the supply chain and increased the competitive landscape for retailers.

The expansion of internet usage has also spurred on the increase of direct-to-consumer shipping, with traditional retailers adding online purchasing, and thousands of dot-com companies springing up during the early 1990s. E-commerce has allowed corporations inexpensive access to a larger market, and significantly decreased inventory and capital costs associated with traditional retail. Figure 1.1 below shows a typical supply chain structure, indicating how e-commerce allows companies cost savings with bypassing one or more nodes in the supply chain.

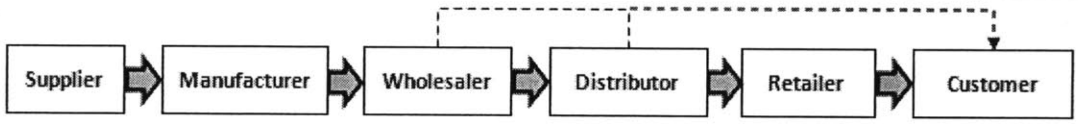

Figure 1.1 General Supply Chain Structure[1].
With the advent of the internet and e-commerce, it has become more and more common to bypass one or more nodes of the supply chain. This reduces complexity and offers cost-savings to the supply chain, which can be used as a competitive advantage.

With the dot-com bust and globalization, and with e-commerce-only companies already cutting out the retail node of the supply chain, the pressure on fulfillment operations and the impact of warehouse management to the bottom line has steadily become more prominent. Innovation in managing the pick and sort processes is especially important at general retailers like Amazon.com, where there is such a wide variety of shape and size to customer orders.

1.1 Amazon.com Incorporated

Amazon.com opened virtual doors to its online bookstore in 1995, as one of the many dot-com companies seeking for a quick presence online. Amazon's vision is to be earth's most customer centric company; to build a place where people can come to find and discover anything they might want to buy online (Amazon.com investor website). In keeping with this vision, over the past decade and a half the product line at Amazon.com has varied to include home appliances, consumer electronics, clothing, and even food.

[1] Adapted from Figure 1.1 in Simchi-Levi (2000), and modified from Bragg (2003)

Employees at Amazon use this focus on the customer not only as a basis for quality fulfillment, but also as a stimulus to motivate waste removal and lean operations. Posters showing empty chairs around a table remind executives and facility managers that the most important attendee, the consumer, is not even present at the meeting. When doing a kaizen or discussing improvement opportunities, the question is frequently asked, "Does the customer care that we are spending money on X?"

1.2 Current State

Amazon.com is built upon three pillars of customer experience: low prices, vast selection, and fast, convenient delivery (Szkutak 2009). In the early days of Amazon, the sortation systems were designed to match the products that were then available (books, CDs, DVDs, etc). With the additional product variety that has come over the past few years, Amazon has begun to redesign sortation systems more robust to product dimension and weight. One of the key initiatives within Amazon.com is to design a flow system towards an ideal that is based around the three pillars of customer experience, as described below:

1) <u>Low price</u> – a system that has high employee productivity and is relatively inexpensive (compared to other options) to install and run.

2) <u>Vast selection</u> – a system that can efficiently handle (with as high productivity and as few errors as possible) the wide variety of product dimension and weight associated with orders containing TEKHO (toys, electronics, kitchen, home, and office).

3) <u>Fast, convenient delivery</u> – a system with a low *shipment cycle time* (interval time from when the first item in an order is picked until the order is packed)

New sortation systems in fulfillment centers in Lexington, KY (LEX1) and in Japan have been developed as the next steps in this initiative. One factor that differentiates these and a few other sites at Amazon.com from traditional warehousing is their use of waveless or continuous flow picking. Amazon.com has been one of the industry pioneers to adopt waveless picking, a process that offers decreased inventory and increased throughput by as much as 35 percent (Bradley 2007). Waveless picking also allows Amazon.com increased productivity for warehouse pickers, which constitutes a majority of the labor directly involved in the fulfillment process. One of the drawbacks of waveless picking, however, is that it can be difficult to manage. Waveless picking requires the use of sophisticated software, and involves a balancing act of constantly managing flow, productivity, and downstream capacity.

Another recent initiative within Amazon.com has been to increase network throughput capacity without opening permanent new sites. One of the challenges Amazon faces is the seasonality of the business. In 2008, 2007, and 2006, Amazon.com recognized 35%, 38%, and 37% of annual revenue during the fourth quarter (Amazon.com 2008). The effect of seasonality on capacity planning is that when new warehouses are opened and staffed to accommodate demand during the peak season, during the remainder of the year the network runs significantly under capacity while still having to pay the fixed costs of the additional warehouses. Some of the ways that Amazon is working to address this issues is through opening temporary, seasonal warehouses, and by encouraging each site to increase throughput capacity during the off-peak months to help accommodate the increased demand during the peak months of the year.

1.3 Problem Statement

The waveless picking, semi-automated sortation system installed in LEX1 in 2007 is seen as the future of Amazon sortation. In order to increase utility before possible rollout of the system to other sites, and to increase throughput capacity in preparation for future peak seasons, Amazon has been researching how to effectively increase throughput without adversely affecting productivity. Since the Amazon Fulfillment Engine (AFE) in LEX1 is a beta-sortation system, and because Amazon is one of the industry leaders in warehouse management and waveless picking, there is not much historical data on managing such a system. The specific issues which are researched and addressed within the scope of this paper are throughput, shipment cycle time, and general management of a waveless picking system.

Although a specific sortation system in an Amazon.com warehouse is used as a case study and basis for the conclusions made in this paper, the principles found herein of managing a continuous flow picking and sortation system can be applied to any warehouse that is currently running or is considering switching to a waveless system.

1.4 Purpose and Deliverables

The purpose of the thesis is to add to the existing knowledge on waveless picking systems, and establish both the benefits and challenges that accompany its adoption. Included in this is an analysis of improvement methodologies, resulting in concrete recommendations which can be used both as an aid in designing a waveless system as well as a roadmap for increasing the utility of a system that is already in place. The thesis also investigates the tradeoff between productivity and cycle time that is inherent in such sortation systems, and attempts to quantify the effect of moving along that continuum.

1.5 Thesis Overview

The thesis is organized into six chapters, as follows:

Chapter 2 gives an overview of warehouse fulfillment operations. Specific focus is made on picking systems, and various types of automated and semi-automated sortation.

Chapter 3 reviews prior literature on warehouse design, pick strategy and optimization, and sortation methodology, and outlays the addition this paper offers to the current general knowledge on warehouse management.

Chapter 4 gives a case study of a waveless picking system in an Amazon.com warehouse, and uses the learnings gained from analysis and experimentation on that system as a basis for recommendations in designing pick scheduling processes.

Chapter 5 provides a comparison of wave-based picking with waveless picking, and considers some of the barriers to adoption of waveless picking that have impeded some warehouse management systems from adoption.

Chapter 6 revisits the results and conclusions obtained through the research, gives a final summary of the benefits and challenges with waveless picking adoption, and provides an overview of future research opportunities for further development of the field of picking theory.

Chapter 2: Overview of Warehouse Fulfillment

Warehouse order fulfillment typically consists of two operations, inbound and outbound. *Inbound* is the process by which product is received from suppliers and stowed in the inventory bins. Effectively managing the inbound operation involves the scheduling of truck arrivals, work release, and labor, in order to balance workflow and reduce congestion. The *outbound* operation includes retrieving the items from inventory for a particular customer order, sorting items into individual customer orders, and packing and shipping product to the customer. The focus on this paper is outbound operations in automated and semi-automated sorters. For further reading and a more comprehensive overview of inbound operations, the reader is referred to Jackson (2005).

2.1 Outbound Process Flow

Outbound operations are more customer-visible than inbound operations, and usually include some or all of the following list of processes:

1) Scheduling – Arranging and timing of dropping orders into the system and scheduling the items in those orders to be picked

2) Pick – Select the items for customer orders out of inventory for further processing

3) Merge and Travel – Conveyance of items from multiple locations in a warehouse to a centralized location for further processing

4) Induct – Remove items from collective packaging (*totes*), and individually inject into automated system

5) Pre-sort – Sort orders into a pre-determined aggregate by size, type, arrival time, or other grouping for later sorting into individual orders

6) Sort – Sort groups of orders into individual order holding cells (*chutes*)

7) Pack – Remove individual orders from chutes, verify contents of customer order, and place in shipping container

8) Label and manifest – Determine shipping method, and apply shipping label

9) Ship-sort – Sort incoming shipping containers by shipping method or location

10) Ship – Process groups sorted by the ship-sort process into the appropriate transportation method

A graphical depiction of the first seven process listed above is shown below. Additional insights and information on the scheduling, picking, and general sorting processes will be discussed later. While inbound process efficiency affects inventory errors, and to some extent picker productivity, outbound efficiency directly affects customer experience with respect to cycle time and shipping errors. This means outbound operations directly impact a customer's view of order quality, and that gains in outbound cycle time, throughput, and quality are immediately visible to the consumer.

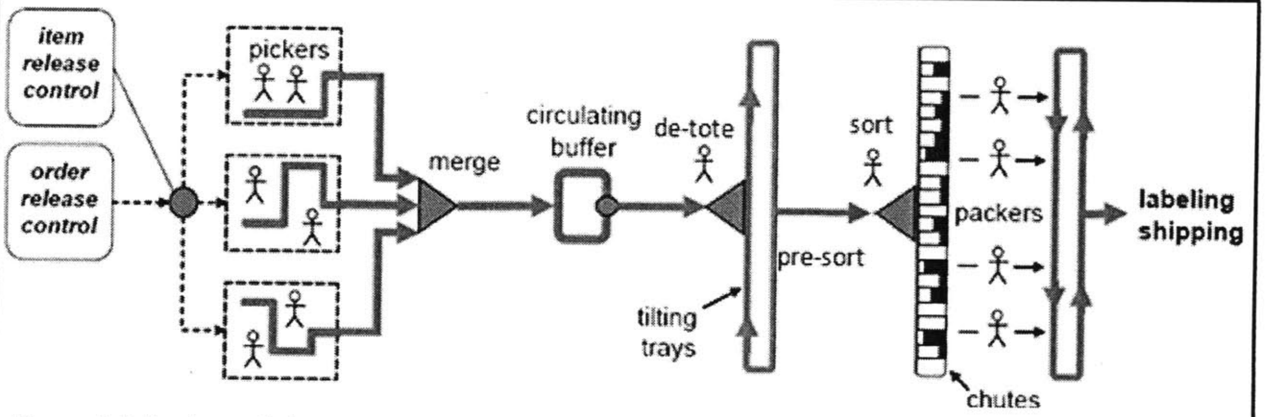

Figure 2.1 Outbound Flow Process in Warehouses with (Semi-)automated Sorters[2].
Fully automated processes may merge the pres-sort and sort into a single, automated process. Buffers shown are indicative of typical placement, but additional buffers may be placed (or removed) from the process depending on the application.

2.2 The Scheduling Process

The scheduling process is crucial because it sets the stage for cycle time and capacity utilization through the rest of the system. Historically, scheduling has primarily been focused on order release control – managing the time that an order is dropped into the system. As will be discussed later, most of the systems currently in place employ wave-based picking, meaning that once an order has been grouped into a wave and dropped into the system, a simple optimization or near-optimization algorithm can be applied to all the items in the wave to determine item scheduling and release control.

An interesting development in complexity comes in scheduling item releases for waveless picking. Unlike wave-based picking, waveless picking does not necessarily require the release of all individual items of an order that has been partially picked into the system. Therefore the scheduling process can now involve not just a determination of the proper time to release an entire order into the system, but also a calculation of when each item within each order must be picked.

Some important factors that are included in many scheduling systems and which can help differentiate between better and worse scheduling algorithms are: customer-promised date of receipt, number of items in an order, item velocity, similarity or distance of items compared to other orders, current location of pickers, throughput capacity, and projected picker walk path. Lodree et al. (2009) discuss the possibility of also including worker safety and ergonomics as a

[2] Adapted from Gallien and Weber (September 2009)

part of the scheduling optimization algorithm, and adapting to maximize for worker productivity given certain limitations on human capability and accounting for cognitive human characteristics and behavior.

2.3 The Picking Process

At a high-level, the picking process is simply a result of the scheduling algorithm employed, and involves just the transfer of product from an inventory storage location (shelf, pallet, rack, etc.) into a carrying tote for transport to the sortation area of the warehouse. At a more detailed level, picking is a strategy which should be chosen to best suit the needs and overall strategy of the corporation.

Making picks from a printed list of needed product *(paper picking)* versus allowing the scheduling software to push picks to a hand-held wireless device held by each picker (*wireless 2-way communication,* as described in Gallien and Weber, September 2009) represents one possible choice which needs to be made. Another could be using *zone picking* to restrict picker movement to pre-determined shelving areas in the warehouse, or utilizing *full-path picking* by restricting pickers to traverse zones but only pick for one downstream processing path.

Additional decisions to be made which could affect picker productivity are whether pickers should travel to the sortation area to manually unload picked merchandise (*pick-unload,* a process which may be better suited to companies unable to invest the capital necessary for a conveyor system, or for larger, non-conveyable items), or if *conveyance* systems should instead be used. Yet another could be allowing pickers to pick any item into a tote together with any other item going to the same processing location (*mixed tote*), versus restricting picks to a single stock keeping unit per tote (*singles),* or even allowing a group of totes to completely contain a finite set of customer orders (*wave picking).* Another option could be *waveless picking,* described in section 2.3.2, where items are randomly assigned to the same tote as other items based more on picker productivity than on completely containing a finite set of orders within a finite number of totes. Table 2.1 below lists a summary of a possible set of decisions affecting the pick process which should be made to align with company vision or strategy.

Base choice	Added Complexity	Potential Gains from Added Complexity	Potential Complications
Paper picking	Wireless 2-way Communication	Real-time flexibility in scheduling	Complexity and infrastructure
Full-path	Zone	Increased productivity	Labor-scheduling complexity
Pick-unload	Conveyance	Increased productivity	Lost product and capital investment
Wave	Waveless	Increased productivity	Loss of control over process and need for extra chute capacity
Singles	Mixed tote	Increased productivity	Added touches

Table 2.1 Pick Process Strategic Decisions

It is clear from the table above that one of the key motivating factors to making process improvements to picking is increased productivity. Since in a typical warehouse picking accounts for over well over 50% of total operating costs, managing picker productivity is a primary concern in designing and running a warehouse operation (Coyle, Bardi, and Langley 1996).

2.3.1 Wave Picking

In the early days of warehousing, single orders were assigned to a single picker and picked as they arrived into the system, in a first-in-first-out type of process (*strict order picking)*. Wave picking was developed in response to the large distances walked by pickers and the increased output volume by direct-to consumer warehousing. Wave or batch picking is performed by allowing a number of orders to accumulate in the system, then strategically choosing a group of them (typically 30-90 for small batches, and up to three times or more that for larger waves). Waves are generally formed based on product density and customer promise, with the end result leading to increased picker productivity. In a comparison of wave and batch picking to strict order, *sequential zone* (where an order is assembled sequentially from zone to zone), and *batch zone* (a combination of batching and zoning) picking, wave and batch picking were observed to be more robust across a wide range of operating conditions, with increased productivity and up to 60% decreased travel time (Peterson 2000). A diagram of wave picking is shown in Figure 2.2 below.

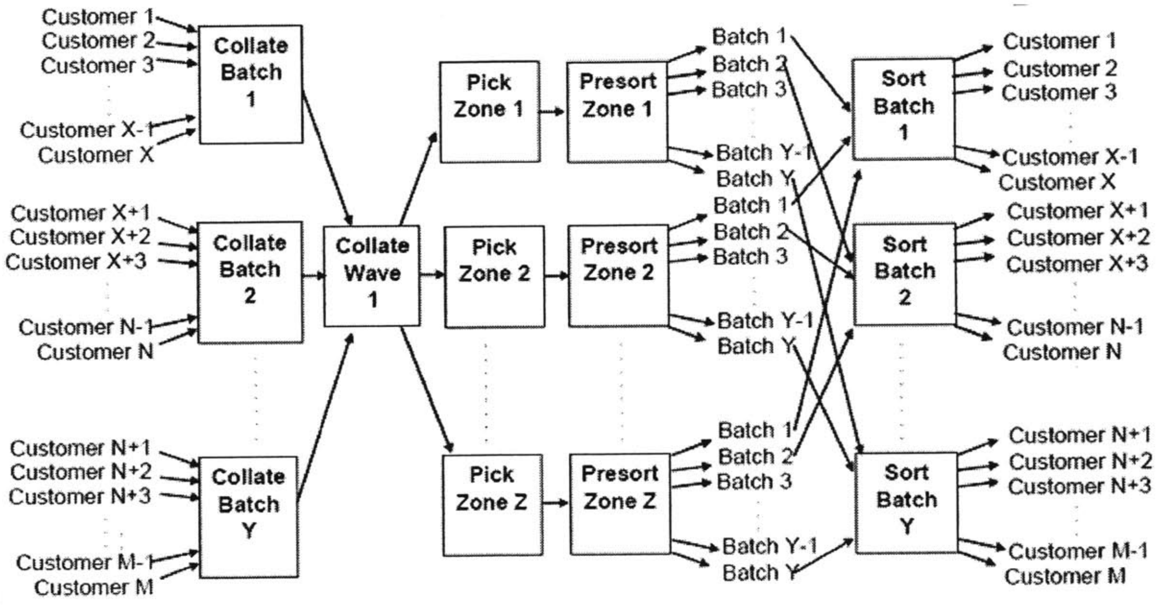

Figure 2.2 The Formation of Waves and Batches[3]

Waves can be thought of, and are often created as, a batch of batches. The above figure shows wave, zone picking of Y batches of various order sizes, as it may be implemented in a typical warehouse.

2.3.2 Waveless Picking

In more recent years, with the amassing knowledge and research on warehouse picking and with the increased sophistication of IT infrastructure, a number of companies have turned to waveless picking for an even greater increase in picker productivity. Waveless picking is a scheduling-intensive process where, rather than releasing an entire order into the system, items are released into the system one at a time. In practice, waveless picking allows for any item to be available as a potential next pick, subject to whatever constraints may be placed on the scheduler to account for system capacity, customer promised ship date, etc. This is in contrast to wave picking where only items within the pre-created wave are available for picking.

Because waveless, or continuous flow, picking does not impose stringent restrictions on which items can be picked, the available pool of orders to pick can be increased dramatically, thus increasing the density of picks along a picker's pick path. This offers the opportunity for increased pick productivity. Two additional implications that this has on the fulfillment system are real-time scheduling and increased pick management.

Real-time scheduling means that the scheduling software can react in real-time to orders that may drop into the system throughout the day. In a wave-based picking system, such orders

[3] Figure 2.2 is adapted from Figure 3.1 in Bragg (2003)

would have to wait until another wave could be created to be picked, regardless of how close the items may be to other items in other waves that pickers may be currently scheduled to pick. In a waveless system, the scheduler can and should be updated to reflect the new orders, and the option is available to have pickers pick items that may be near other items which are currently scheduled to be picked, thus potentially increasing picker productivity.

Increased management methodologies may be necessary to handle the additional flexibility offered by waveless picking. For example, without any special restrictions or management imposed on the picking system, it is possible that an order may never be completely fulfilled. Consider the scenario where an order drops into a waveless picking system that has been configured to optimize based solely on average density of picks for all pick paths. Let one of the items in the example order be a high-velocity item, and another be a low-velocity item that happens to be stored in a corner of the warehouse that happens to be surrounded by other low-velocity items. This assumption is not unrealistic if the warehouse is utilizing random stow of inventory. It is then possible that the first item mentioned above gets picked simultaneously with others of the same item for other customer orders. The order can now be considered "open", or partially but not completely picked. If the software is optimizing for productivity, it is possible that a significant amount of time may pass before a picker is pulled to the area of the warehouse containing the second item. See Figure 2.3 below for a graphical representation of this case.

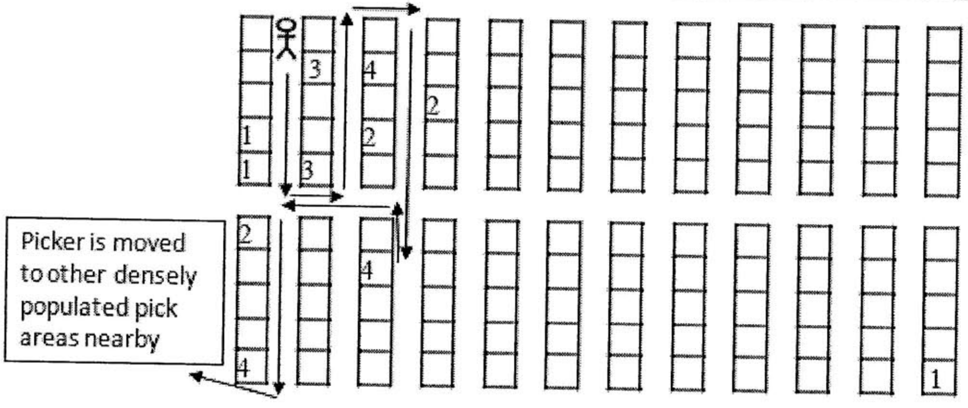
Picker is moved to other densely populated pick areas nearby
3
4
2
1
2
1
3
2
4
4
1

Figure 2.3 Waveless Picking and Scheduling by Picker Productivity
The above figure depicts items for four different orders. The location of an item is indicated by the number of the order it belongs to. In this example, order 1 has a low-velocity item at the far corner of the warehouse that may not be picked for some time.

Although it would be irrational to design such a system that is based solely on productivity without accounting for customer promise and other factors, this example indicates the added complexity of managing the scheduling of each individual pick, and the potential loss of productivity that may occur by chasing down items which were not picked in a timely manner through the normal scheduling process. More detail on the potential advantages and disadvantages of waveless picking compared to wave-based picking are discussed in Chapter 4.

2.4 The (Semi-) Automated Sortation Process

Regardless of whether a warehouse employs wave or waveless picking, as long as customer orders contain multiple items, and items are small enough to fit two or more per tote, totes will arrive to the sortation process mixed with multiple orders. As previously mentioned, the primary difference between these two picking strategies is merely that a defined, finite number of totes completely contain a finite number of orders in wave picking, while in waveless picking there is no predetermined set of totes containing a complete set of customer orders. After totes arrive at their destination, items are removed one at a time from the containing totes, scanned to verify location within the fulfillment center, and then inducted into the sortation process. Items must then be sorted into their respective orders for further processing.

Final sorting is done by placing or dropping items into individual chutes, where they wait while the entire order accumulates. Figure 2.4 below shows a graphical representation of how chutes are utilized, and how that relationship is largely dependent upon the pick cycle time, or time from the first pick in a customer order to the last pick in that same customer order. Any number of items may be picked and sorted to the same chute, but all multi-unit orders will have the same governing characteristics relating to sorting and chute usage as indicated below.

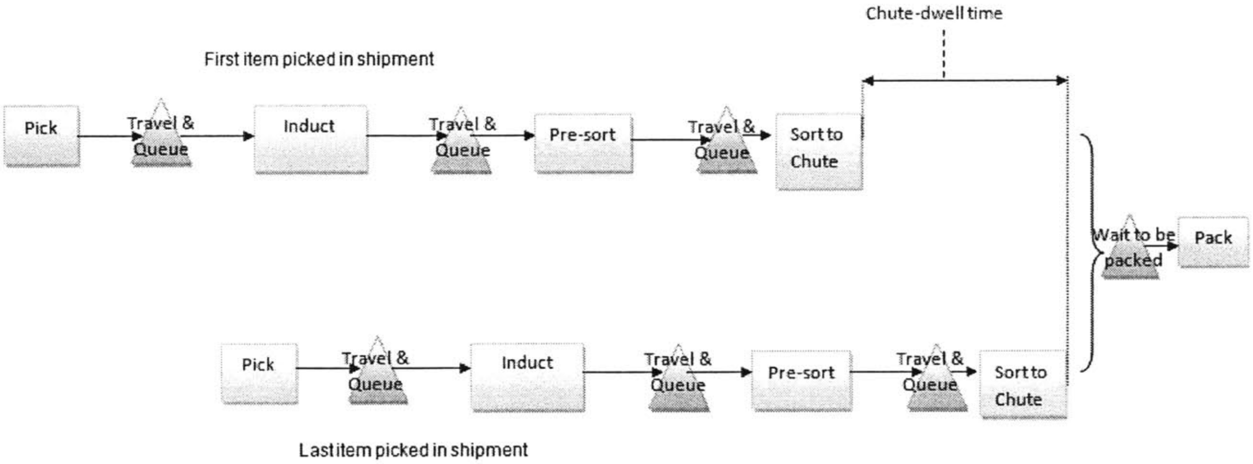

Figure 2.4 Shipment Lifecycle and Chute-dwell Time

A fully automated sort process will henceforth be defined to mean that once items have been inducted into the system, they are not touched again by an operator until an entire order has congregated into a chute, ready for further processing. The entire chute assignment and sorting process is performed automatically by the sorter.

A semi-automated sort process will be defined to indicate a process where, once items have been inducted into the system, they are assigned to a co-located group of chutes by an automated pre-sorter, then final sortation into individual chutes is performed manually. Another, similar example of semi-automated sorting which can be employed only in a wave-based system, is if, rather than inducting the items onto the system individually, the totes are automatically pre-sorted into batches and sent to different lanes for final, manual sorting into individual chutes.

2.4.1 Sorter Capacity

In a system with an automated sorter or pre-sorter, the largest capital investment and most difficult component to make changes to is usually the sorter itself. Conveyor speeds across the entire system are chosen to appropriately match or scale to the sorter speed. Conveyor merges are timed and downstream high-speed conveyors matched based on the sorter speed. Additionally, the sorter represents a large design and engineering investment with many interdependencies and intricacies that are not well inclined to changes in speed or timing. With all these considerations, a sorter is generally designed or purchased with a given capacity, which represents the maximum threshold throughput capacity for the entire system. This also means that any throughput limitations which are observed in practice that are below the sorter capacity can be viewed as system inefficiencies which the operations team can affront if additional throughput capacity is needed. Even sorters that can run with peak utilization periods of 100% may have a much reduced effective utilization closer to 60% or so, depending on the picking and sorting strategy used (Perry 2007).

One caveat to the idea that the sorter capacity is fixed, needs to be mentioned. Actual sorter capacity depends on the implementation. If the sorter contains *tilt-trays*, or physically separate carrying compartments for individual items, with a combined width and spacing of w, and can run at a maximum speed s, then the theoretical sorter capacity R is a finite quantity described by the equation:

$$R = s / w$$

Equation 2.1 Theoretical Sorter Capacity

Another type of sorter implementation is a *kick-out sorter,* where the sorter does not itself contain physical tilt-trays, but operates by placing items either directly on a conveyor or placing items in small bins which are then placed on a conveyor. The items are then "*kicked out*", or pushed off of the conveyor by automated shoes, and thus individually sorted to final sort or to a specific chute, depending on system design. The theoretical capacity remains the same as described above for tilt tray, but w becomes a variable which could be changed much more easily by changing the item container size, with a corresponding change in throughput capacity.

2.4.2 Operator Capacity

If the sorter is not able to be run at or near the theoretical capacity of the sorter, one possible explanation is that operators are unable to complete the tasks of induct, rebin (manual sort), packing, or downstream processing quickly enough. In such a situation, whether the capacity limit is upstream or downstream, respectively corresponding to starving or blocking of the sorter, can be determined through a capacity analysis of the system.

One complicating factor in production systems that include a series of manual operations with emphasis on worker productivity, such as in warehouse fulfillment, is the tendency to find dual, or floating, bottlenecks. By attempting to minimize the number of workers for a maximum productivity per operator, the available capacity at every resource equals the required capacity at every resource (Hurley 1998). The effect of this policy of waste removal is that it removes all protective capacity from the line, causing all the stations that are closely matched in capacity to unnecessarily starve and block each other, further limiting throughput. Protective capacity in non-bottlenecks may need to be as high as 20% excess capacity in order to reduce "shiftiness", or the tendency of the bottleneck to shift between stations, to a reasonable level (Patterson 2002). This implies that in order to run such a system at maximum throughput, the bottleneck might best be chosen strategically, with all other stations operating with 20+% excess capacity (in the form of excess operators). Otherwise, multiple stations will appear as the bottleneck at various times, throughput will be further reduced below the capacity of any individual potential bottleneck, and operations may not address any individual bottleneck since it would just shift the bottleneck to another operation of similar capacity, and give no immediate increase in throughput. Adding protective capacity, and strategic placement of the protective capacity, can optimally utilize the bottleneck with minimal resources (Craighead 2001).

2.4.3 Chute Capacity

Another possible root cause for decreased throughput within a sortation system is chute capacity. Chutes are holding cells for individual orders, where items wait while the rest of the contents of each order accumulate. Chute capacity, or the number of usable chutes, is usually a fixed quantity that cannot be easily adjusted.

An immediate impact of a limited chute capacity is on the allowable work in process at pick. For example, a wave consisting of 500 customer orders will simply not fit if the chute capacity is limited to 400 chutes. Thus, chute capacity can force operations to reduce upstream productivity, and possible capacity of the sorter, in order to avoid overflowing the chutes. The situation is only complicated with overlapping waves, where subsequent waves begin to be picked before earlier waves are completed. In such a system, a chute capacity of 400 chutes may limit wave sizes to 300 or even 200 chutes, depending on the nature and timing of the overlap.

Chute capacity can be even more subtle yet impactful in a waveless picking system, where waves are not created in such a concretely defined way. Obviously, this puts a constraint on a waveless system that, although any item in any order is available to be picked at any given time, *closers* (the final, or closing, pick for a give order) must be made frequently enough that occupied chutes can be turned over to avoid overflowing chutes. Or conversely, *openers* (opening picks for a given order) can only be made on average with the same frequency as closers. As an example, consider a scenario where, for a period of 10 consecutive hours, opening picks are very densely populated, and the scheduling system picks openers at a rate that is 10% higher than the rate at which closers are picked. If the closer pick rate were 100 per hour, than at the end of ten hours, 100 more chutes would be occupied than at the beginning of the time period. In addition to managing the pick ratio of openers to closers, the scheduler must now place an upper bound to the quantity of partially picked orders. Even in a static system, if only 400 chutes are available, but the scheduler allows 500 orders to be partially picked, it is likely that the chutes will overflow.

Even with a scheduling system that operates under the balancing requirements listed above, some variation in chute utilization will be present due to the variation in tote travel times from pick to induct, and the variability in travel time from induct to item arrival at the chute. Thus, limiting the system mentioned above to 400 partially picked orders, or even 390, may not be sufficient to avoid *chute overflow*, the phenomenon when there are not any empty chutes to accept new openers that flow down from the sorter.

Any of the above scenarios, for both waveless and wave picking, could result in chute overflow. Because chutes are limited, and the system will not sort items to chutes containing another order, the actual result is that congestion will build upstream as the system waits for items that will free the utilized chutes. This increased congestion and recirculation of items that are unable to be sorted to an open chute due to limited chute capacity, will cause a reduction in throughput at the sorter itself. In an extreme case of congestion, the very items needed to free the filled chutes may be unable to reach the sorter because of the same congestion (Gallien, August 2009). Such a situation has been coined "gridlock" (Johnson 1994), and is a frequent result of chute overflow.

2.5 Crisplant

One widely used example of a fully-automated sorter is a Crisplant tilt-tray sortation system. The Crisplant system is a high-capacity induct system similar to those analyzed in Johnson and Meller (2002). The Crisplant system utilizes manual induction labor, automatic sort into individual orders, and manual order packing. As such, the Crisplant tilt-tray sortation system represents two interesting cases of throughput limitation. In their description of such sorting systems, Johnson and Meller affirm that induct is usually the sorting system bottleneck:

> "And although it is true that the throughput of each subsystem is dependent on the labor assigned to that activity, only induction is also limited by the investment and configuration of the hardware. Thus, it is rare to find a system where the induction process is not the bottleneck of the sorting system. In practice, this assumption is supported by noting that the picking process usually works ahead of the induction area by staging cases in front of the sorter, and the packing area includes a mechanism to ensure that packing never blocks the sorter" (Johnson and Meller 2002).

Obviously this indicates a system that is throughput limited by operator capacity. In practice, operators try to maximize upstream productivity by allowing high chute utilization, and thus increasing the pick path density for the pickers. This can sometimes increase system congestion to the point where inductors are blocked, and unable to work at full capacity. In such a scenario, the bottleneck has effectively, albeit by choice, been switched to be limited by the chute capacity. This idea has been supported through observations made where, occasionally throughout the day, inductors were requested by operations to stop inducting and help clear completed chutes in order to decrease system congestion for a period of time (Campbellsville 2009).

2.5.1 Focus on Picking

One of the major benefits of automated systems in general is the improvement in labor productivity in the pick process, which as was previously mentioned, usually accounts for over 50% of the total warehouse operating costs (Russell 2003). The Crisplant system is no different, and can therefore be managed for optimizing the gains which are achieved in picking. Thus, the frequency for congestion and chute capacity limitation on system throughput as mentioned above would not be an unreasonable assumption. Amazon.com works to extend the utility of its Crisplant systems even further by utilizing continuous flow picking to feed its Crisplant-run warehouses.

The focus on the upstream picking process is so strong that downstream process efficiency and productivity is somewhat sacrificed within Crisplant. An example of this is the effect of system setup on pack productivity. The tilt-tray system, which allows for increased pick productivity, has a significant footprint in the warehouse. Since final sort is automated, chutes must span the same travel footprint that the tilt-tray system spans in order to allow the trays to tilt items into the chutes. For example, consider that higher chute capacity means a higher quantity of partially-picked-orders, which means higher picker productivity. This means more tilt trays, and a larger circulation for the tilt-tray system. This in turn requires a larger footprint, which forces packers to travel more between the packing out of shipments, which reduces pack productivity significantly below pack rates for wave or other standard processes within the Amazon network.

2.5.2 Management Methodology

Crisplant facilities are used primarily for the gain in productivity. As a result, within Amazon.com, Crisplant facilities are run by maximizing the number of partially-picked orders without exceeding chute capacity. This increases picker productivity and fully utilizes chute capacity on the sorter. Chute availability depends not only on the number of chutes with orders still accumulating *(incomplete chutes),* but also on the ability of the packers to clear out completed chutes *(packable chutes).* Thus, actual chute availability may go up or down depending on the pack labor availability for downstream processing (thus affecting throughput capacity for the day), but the capacity utilization of incomplete chutes only remains fairly constant. Induct productivity is also a key metric in the Amazon.com Crisplant facilities, and inductors receive real-time feedback on current productivity rates as well as being given enough pre-station queuing to ensure that each worker is never starved for work.

The above factors lead to significant efforts in upstream processing to ensure that totes arrive to sortation in such a way that minimizes the time from first item arrival to the chute until the time of the last item arrival *(chute-dwell time)*, by allowing totes with higher concentrations of closers onto the system before other totes.

The result of this focus on productivity and chute utilization is an increase in queuing inventory. Queue times from when an order arrives to the Crisplant area until it is sorted into a chute can be as long as 2-4 hours of pure wait time (not including order accumulation time). Congestion is high and the probability of gridlock is consequently high as well.

2.6 Amazon Fulfillment Engine

The Amazon Fulfillment Engine (AFE) is a new breed of sortation systems within the Amazon.com network that began operation in 2007. AFE employs manual induct to feed an automated pre-sort process. Manual final-sortation into individual chutes completes the sortation process, followed by a standard pack and ship process.

2.6.1 Comparison to Crisplant

Amazon Fulfillment Engine product flow is very similar to Crisplant, with three major differences in design: 1) AFE uses a kick-out sortation system off of a sort-conveyor (as described in Section 2.4.1) rather than a hard-limit capacity tilt-tray system, 2) AFE was designed specifically to transport and sort larger items than can fit within the Crisplant conveyance and tilt-tray system rather than with the primary focus of maximizing the induct capacity per inductor, 3) chute capacity is much higher and chutes are more densely populated than in Crisplant.

Presumably these changes were made to address the major limitations of the Crisplant system within the Amazon.com network. First off, the tilt-trays on Crisplant are weight-limited by what can be carried on them. Additionally, the size of the tilt tray and the downstream chute processing was designed for Amazon.com's original size and shape of product (BMVD: books, music, video, and DVD), and is inflexible to the wider product variety which Amazon.com now offers. Secondly, as mentioned in Section 2.5.1, the way that the Crisplant sortation system is run to maximize picker productivity leaves the system prone to high chute capacity utilization and a reduction in productivity in some of the sort processes within the warehouse. By allowing for conveyance kick-out sorting in the Amazon Fulfillment Engine, trays can be chosen and even adjusted in the future to accommodate product size. Increasing chute capacity and density mitigates the issue of chute utilization by increasing total chute capacity, potentially increases packing labor productivity, and helps reduce blocking of upstream processes.

2.6.2 Warehouse and Management Organization

The LEX1 warehouse where the Amazon Fulfillment Engine is contained consists of three connected buildings, A, B, and C, which connect linearly in the same order. Building A contains some inventory shelving and replenishment shelving (where additional items are stacked in locations which are not readily pickable, for replenishment when the pickable locations for that item are depleted). Building B houses the majority of inventory shelving and inbound operations. Building C contains outbound processing and shipping, with the Amazon Fulfillment Engine being located at the far end away from building B and A. Processing for other *process paths* (distinct destinations and processing methods such as: wave processing, singles, specialized processing for DVDs, CDs, etc) is also located in Building C. A high-level diagram of the AFE process itself (modified to protect Amazon.com confidential information) is shown in Figure 2.5.

The management of the Amazon Fulfillment Engine is very simply divided into two teams, the picking team and the sortation team. Each team has separate frontline managers that report to the same outbound operations manager for each shift. The picking team is responsible for the picking and transportation of picked items as well as the balancing of pick labor to ensure that each downstream processing location receives a consistent stream of work. For AFE, the pick team also holds responsibility over managing the system-governing parameters that determine the behavior and performance of the AFE system (more detail on these parameters is given in Chapter 4). The sortation team is responsible for the induct, sort, pack, and ship processes. Software development and maintenance is split into three groups: picking, sorting, and shipping. Each software development group is remotely located, and works with the picking and sortation teams to develop and manage the software to the facilities needs.

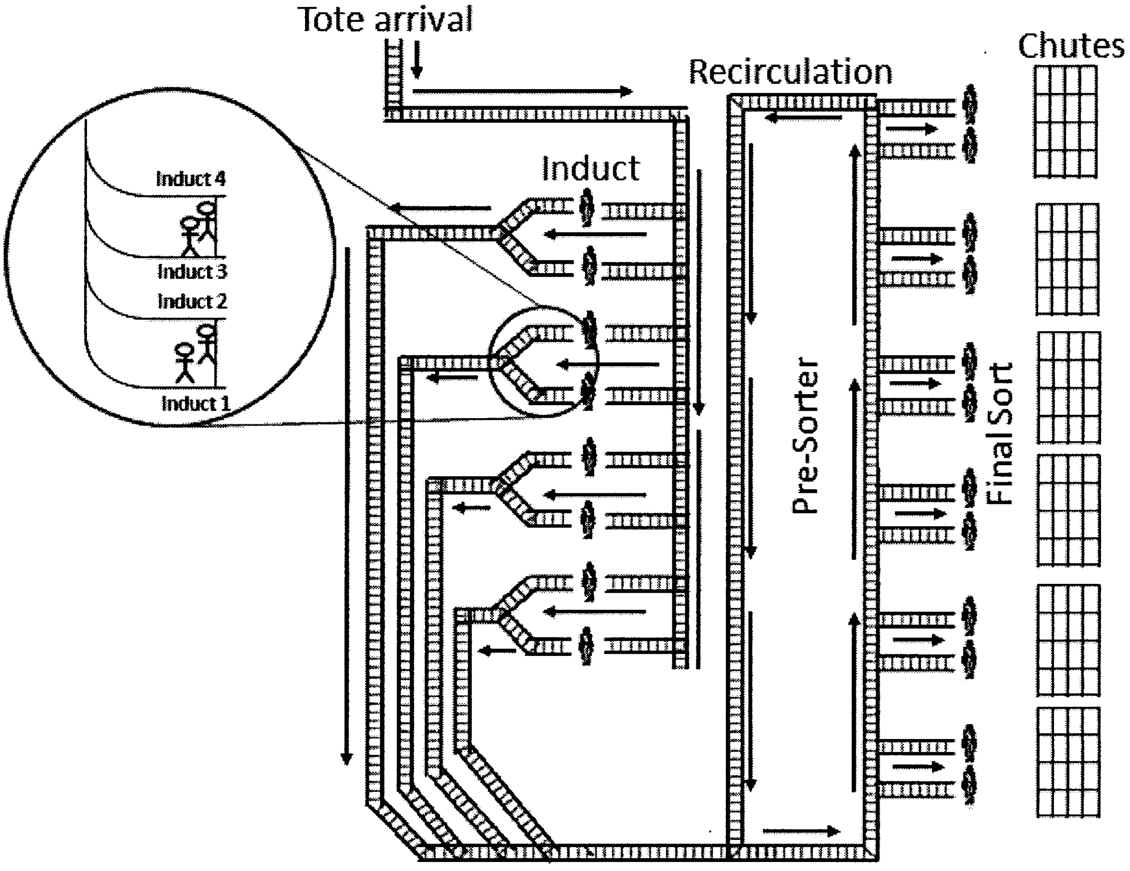

Figure 2.5 Amazon Fulfillment Engine Product Flow Diagram

2.7 Summary

This chapter provided an introduction to warehouse fulfillment operations, and offered an overview of the major process components involved in automatic sortation in a retail or e-retail warehouse. The key concepts from this chapter are as follows:

1) Product flow within a warehouse fulfillment system is characterized by scheduling of picks, picking, sorting, and packing out a completed customer order

2) Wave picking is the process of batching together customer orders, then picking the entire wave as one large order.

3) Waveless picking is a relatively new fulfillment process that adds complexity and real-time flexibility to the pick process, with potential gains in throughput and productivity over wave picking.

4) Automatic sorters can fall into two categories, fully automated and semi-automated. Each type of sorter accumulates orders in chutes, and is operationally limited in throughput and productivity both by sorter capacity as well as chute capacity.

5) The Amazon Fulfillment Engine represents a unique sortation system that utilizes waveless picking, high physical density of chutes, and a kick-out conveyance sorter to aid in the fulfillment process.

Chapter 3: Literature Review

This chapter contains a review of papers focused on the development of picking and sortation theory. As this paper is restricted in scope to automated and semi-automated sortation systems, the majority of the literature surveyed in this chapter is relevant to such systems.

One group of papers analyzes how to optimally design a warehouse to maximize productivity and throughput. Gue challenges the standard notion of parallel shelving aisles for pick labor productivity with shelving in a slightly curved "V", asserting that such V-shaped cross-aisles can reduce travel distance by 8 percent to 10 percent (Gue 2009). Pohl et al. (2009) research three common pick shelving layouts, and analytically determine some decision rules for choosing a warehouse shelving design and layout. Kevan (2004) and Friedman (2006) separately explore the efficiencies gained through implementation of a high-tech voice-recognition system in warehouse picking, freeing the eyes and hands of pickers and increasing productivity and safety. Baker and Canessa (2009) take a step back with a more holistic approach, and analyze each step in warehouse design individually along with its effect on the entire system. Bragg (2003) performs a cost-based analysis to some warehouse designs employed at Amazon.com.

A second set of papers recount the development of pick operations, analyzing various schemes and optimization methods for the same. Ackerman (1990) describes three picking strategies: strict order (picker picks one order at a time), batch (picker assigned more than one order at a time), and zone (picker picks only items in his/her assigned zone). Peterson (2000) contributes to this breakdown of picking strategy, and performs simulation modeling and side-by-side comparison of strict order, batch, sequential zone, batch zone, and wave picking in mail order companies. He concludes that wave and batch picking were robust across operating conditions, and that sequential zone and batch zone picking are ineffective. Johnson and Lofgren (1994) research the usefulness of simulation modeling in wave picking, resulting in a greater than 60% increase in the cost of total warehouse operations. Gademann et al. (2001) research an optimization strategy using a branch-and-bound algorithm for batch creation that minimizes the maximum lead time on any of the batches. Owyong and Yih (2006) consider a "push algorithm" in wave-based picking as a heuristic to modify pick lists in order to minimize pick cycle time and thus chute dwell time. Using empirical data, Hinojosa (2006) extols the picking and throughput increases (up to 20% and 35% respectively) which can be attained from switching pick operations from wave-based to waveless. Bradley (2007) mirrors the praise of Hinojosa, while considering the capital requirements and software necessities in making such a change. Parks (2008) discusses waveless picking in an American Eagle Outfitters distribution center as an upstream process feeding two different sorters. Gallien and Weber (August 2009) perform one of the first direct comparisons between wave picking and waveless picking, showing that in the case simulations studied, waveless picking had equal or larger throughput with lower probability of gridlock in all optimal scenarios than any wave-based policy considered.

Another set of papers explores the particular use and efficiencies of automated sorters, researching both the impact to sortation efficiency as well as the impact of sortation methods to picker productivity and strategy. Kator (2007) examines how Fender Musical Instruments installed an automated sorter to achieve 30% increase in throughput, while improving order accuracy and without adding labor. Johnson (1998) compares two sorting schemes, fixed priority and next available, to determine sorting strategy impact on sorter performance. Bozer et al. (1988) performed a simulation study to research sorting strategies, concluding that assigning orders to the first available lane outperformed more elaborate assignment methods for the sorter type used in the case study. In Choe (1990) and Choe et al. (1992), a specific analysis is performed on the impact of sortation systems to the pick process, and the relationship between pick and sort is defined using queuing models. Gallien and Weber (August 2009) develop a model to describe probability of gridlock on a split-case sorter given certain assumptions on picking policy and pack labor. Parks (2008) examines the result of installing two separate sort processes, a Dematic high speed sorter and a pick-to-voice system, in order to better align the sort process with products and upstream and downstream processing.

The addition that this paper makes to the existing literature is to further examine the adoption requirements for waveless picking, and to offer a case-study analysis of management best practices and potential pitfalls for adopters of waveless picking who are or will be in the early stage of implementing the waveless scheduling process.

Chapter 4: A Case Study, Waveless Picking and AFE

A brief introduction to the function of the Amazon Fulfillment Engine (AFE) was presented in Chapter 2. In order to review management policy of AFE and the notable effects of waveless picking there, a more detailed discussion of the AFE system functionality is needed. In this chapter, a detailed analysis of AFE is made, followed by experiment setup and results leading to further discussion on the impact of waveless picking on a fulfillment system.

4.1 Management of AFE

Amazon labor is divided by function rather than by process path. (Many warehouses employ multiple process paths to most efficiently handle larger versus smaller order, very specifically sized orders, etc.) Thus, for example, all picking is managed together, separately from each of the individual sortation paths that the pickers are feeding. Therefore, AFE product flow is placed within the jurisdiction of an autonomous picking team. As previously discussed, one of the main drivers for adopting waveless picking is the potential gain in pick labor productivity. By placing control of flow in picking, management emphasizes the importance of pick productivity to the AFE process path.

4.1.1 Waveless Picking in AFE

Consistent with the focus on pick labor productivity within the Amazon Fulfillment Engine, waveless pick scheduling is done with heuristics aimed at maximizing picker productivity. Pickers are assigned to work within a pick zone that has a significant pick volume and density to warrant a picker assignment. Picks are then scheduled using the optimization-focused heuristics currently in place, with the purpose of maximizing productivity. Small additional constraints are in place (such as picking openers and closers at the same rate to avoid a step function change in downstream chute utilization, parameters to encourage picking a pre-specified ratio of work due to be shipped within the next 24, 48, 72, or more hours, and so forth), but for all intents and purposes the goal of the scheduler is to maximize productivity while allowing the system to remain in a stable state with constant chute utilization.

A major productivity-limiting factor in place that is used as a management lever to control the scheduler and system behavior is the restriction on the number of partially picked shipments, which can be referred to as the *shipment limiter*. The shipment limiter is theoretically the maximum number of openers that have been picked to which the corresponding closer has not yet been picked. By enforcing the shipment limiter, actual pick density is reduced since some opening picks are removed from the list of possible picks in order to allow closers to be picked first. See Figure 4.1 below for a sample sequence of events in a system constrained by a shipment limiter. In the example in this figure, order *m* is not picked due to the violation of the shipment limiter which would occur if it were picked before completing another order first. This occurs despite the increase in productivity which would occur if it were picked at the same time as the opener for order *k*.

Obviously, in a system that has workers that pick at various rates with random variation in instantaneous pick rate, the exact number of open shipments will vary from the scheduled plan simply due to variation in the timing of the picks. The shipment limiter can be used as a target rather than a maximum threshold, with the purpose of maximizing pick density and therefore picker productivity, within the limits of a constraint on chute capacity. In such a system implementation, as is used within the Amazon Fulfillment Engine, the actual number of partially picked orders at any instant in time will vary randomly around the shipment limiter setting.

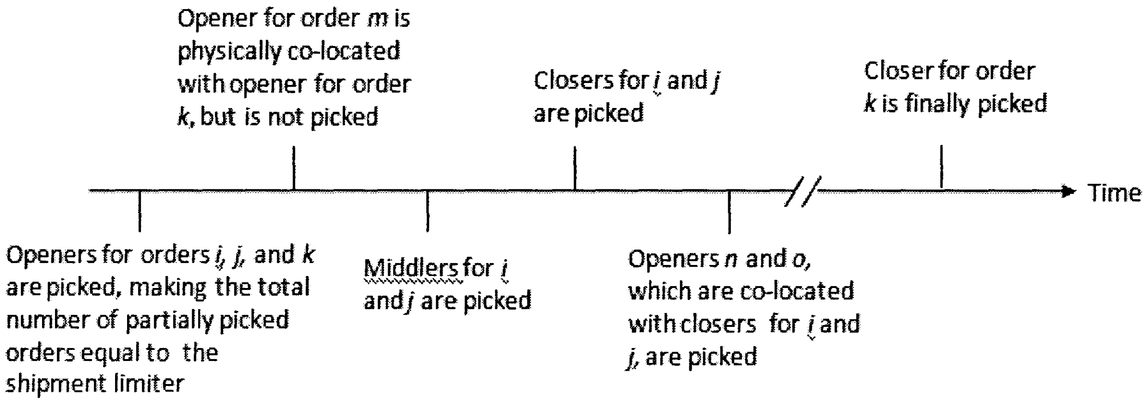
Opener for order *m* is physically co-located with opener for order *k*, but is not picked
Closers for *i* and *j* are picked
Closer for order *k* is finally picked
Time
Openers for orders *i*, *j*, and *k* are picked, making the total number of partially picked orders equal to the shipment limiter
Middlers for *i* and *j* are picked
Openers *n* and *o*, which are co-located with closers for *i* and *j*, are picked

Figure 4.1 Potential Shipment Limiter Effect on Cycle Time and Productivity

Figure 4.1 shows a potential cycle time loss on order *k* due to item placement together with the productivity loss from not picking items in order *m*.

The second productivity-limiting factor within the Amazon Fulfillment Engine, as described in section 2.3.2 of this paper, is a method to limit how long a pick can dwell before it is forced to be picked. While the shipment limiter may allow only a given number, say 500, orders to be partially picked, it does not inherently enforce which of the currently open 500 must be completely picked before a new opener can be picked. Thus, without this additional mechanism in place, orders 1-100 could potentially remain unpicked indefinitely, while orders 200-500 are continually turned over. AFE uses two parameters to mitigate this. The first, which can be termed the *lateness window*, calculates when a pick must be made in order to meet customer promised ship date. If a pick dwells past this calculated time, it is flagged as "late", and is scheduled to be picked immediately, thus reducing picker productivity and bypassing the scheduling algorithm based on pick density. The second, which can be called the *demand window*, is an attempt to enforce a pick cycle time, and can be thought of as corresponding to the average pick cycle time across all shipments. By allowing some flexibility in individual pick cycle times, the scheduler can attempt to manage the impact on pick productivity of potentially chasing items which violate the demand window.

4.1.2 Chute Utilization in AFE

Chute utilization, as mentioned in section 2.4.3, can turn into a limiting factor for an automated sorter, depending on the management method for the surrounding processes. As such, it is common in automated warehouses to perform real-time tracking of chute usage as part of a management feedback loop. Since increasing chute capacity is usually not an option, managing it appropriately is a necessity (Maloney, 2004). Figure 4.2 below shows an example tracking graph from the Amazon Fulfillment Engine, that management might use to ensure that the random variations caused from attempting to match to the shipment limiter are not sufficient to cause chute overflow in packable and incomplete chutes.

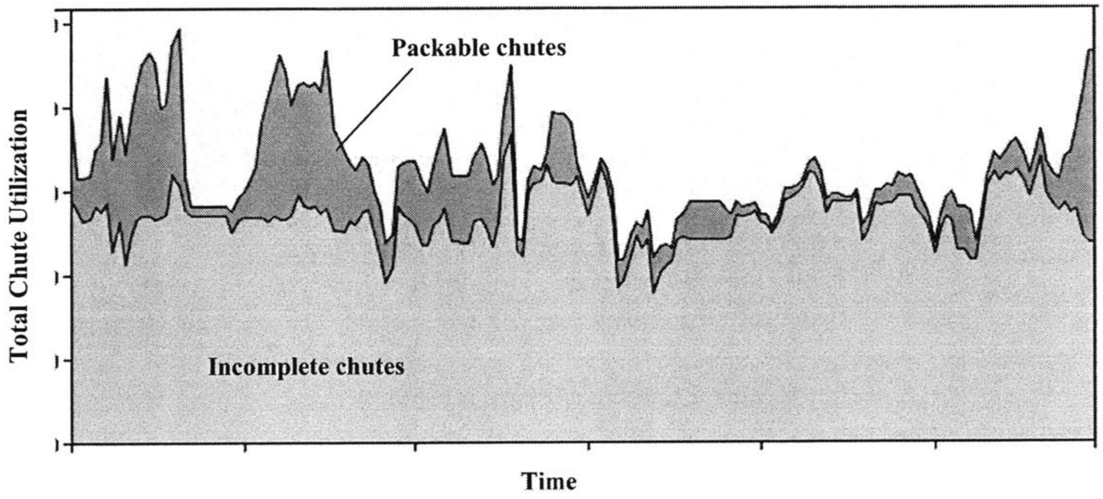

Figure 4.2 Chute Utilization Tracking in AFE

One final, notable feature of the chute setup in AFE is the distinction and separation of accumulation chutes from packable chutes. In many other sortation systems, a chute is considered occupied once an opener is assigned and dropped in to it, and is not free again until a packer packs out the order contained in the chute. In AFE, every chute in which an order accumulates has a corresponding but physically distinct chute where the order can drop to while it awaits packing. The effect of such a system has many subtleties which will be addressed later in this paper, but ultimately this increases chute capacity without increasing walking distance for rebin and pack workers to reach the additional chutes.

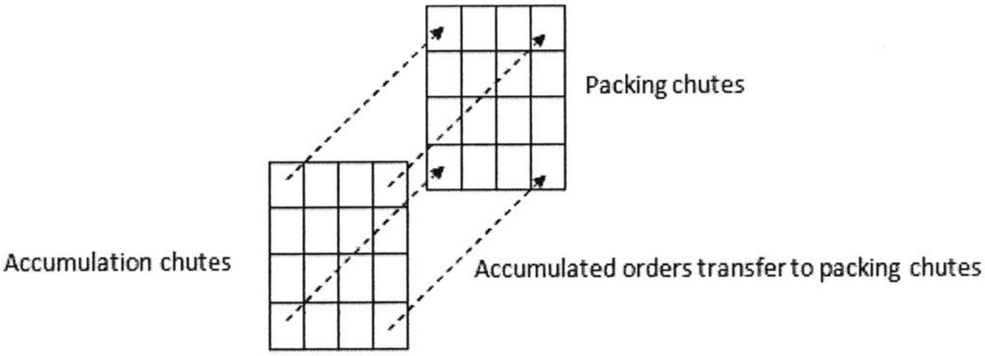

Figure 4.3 Accumulation and Packing Chutes in AFE

4.2 System Functionality and Control

While increasing throughput can be a lofty goal, there is always the question of how such an increase is to be achieved. More obvious (and costly) answers tend to be adding processing power, increasing workforce size, and expanding facilities. However, if resources like space and money are limited, how might throughput be increased? Lean advocates ascertain that applying lean principles with rigor and paying attention to the finer points of a lean/six sigma program will increase throughput without affecting productivity adversely (Jutras 2009). According to the queuing formula shown below in Equation 4.1, cycle time, arrival rate, and work in process (WIP = inventory in the system) are all related in a quantifiable way (Little 1961).

$$L = \lambda W$$

Where: L = expected number of units in the system (WIP)
W = expected time spent by a unit in the system (Cycle Time)
λ = expected processing rate for items in the system (Throughput)

Equation 4.1 Queuing Formula for WIP

Rearranging Equation 4.1, it is obvious that throughput varies linearly with WIP, and inversely with cycle time. This reveals that an increase in throughput can be achieved as a result of lean efforts to reduce cycle time. This relationship emphasizes the importance of cycle time not only to lead time and customer promised ship date, but also on throughput.

4.2.1 Cycle Time Analysis

Three specific concerns about cycle time ought to be addressed as a primary move to determine and understand cycle time effects within a queuing system. First, what factors contribute most significantly to long cycle times? Second, how does order size affect these same factors? And finally, what causal relationships can be deduced for the contributing factors in question?

4.2.1.1 Contributing Factors to Long Cycle Times

Figure 2.3 gave a sample life-cycle for customer orders within a warehouse. A similar analysis for the Amazon Fulfillment Engine confirms that the vast majority of shipment cycle time is contained within the chute-dwell time while an order is accumulating. Not only is greater than 50% of shipment cycle time spent in this area, but the next longest contributing factor to cycle time accounted for less than half the impact of order-accumulation. A study was performed wherein over 400,000 items comprising over 125,000 orders over the course of seven days were tracked through the AFE system to determine processing and wait times for all processing from pick to pack. Table 4.1 below gives a sample distribution of cycle and processing times indicative of the actual results obtained from the study. Refer to Figure 2.3 for a graphical representation of how each process listed in Table 4.1 fits within the item and shipment lifecycle.

Process	% of total shipment cycle time
Tote travel	8%
Induct queue	15%
Tray travel to Sort	10%
Chute-dwell	60%
Wait to be packed	7%

Table 4.1 Sample Distribution of Processing Times

These results help to confirm that chute-dwell time is a key issue limiting both cycle time and throughput in order fulfillment centers of this type.

4.2.1.2 Order Size Effect on Cycle Time Factors

A second issue of concern is how order size may affect chute-dwell time, which has been ascertained to be the largest contributing factor to long cycle times. It can reasonably be expected that chute-dwell time will increase with larger order sizes. Figure 4.4 below confirms this general trend within the AFE system.

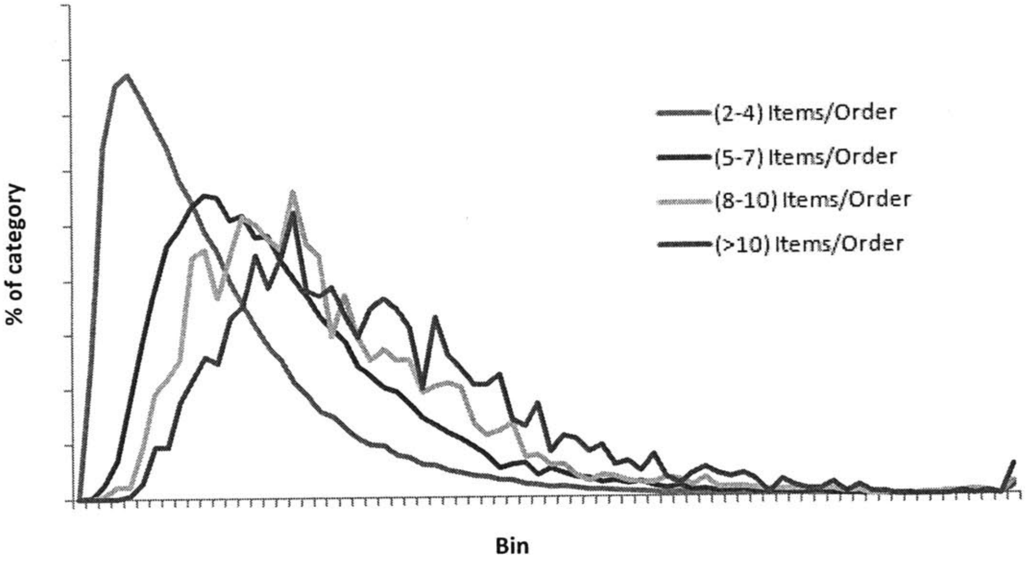

Figure 4.4 Chute-dwell Time Histogram by Order Size

The significance of this is threefold. First, it can be expected that even working optimally, larger orders should take longer to accumulate than smaller orders. Second, larger orders have a much larger mean chute-dwell time, and thus provide a larger opportunity for cycle time improvement for individual shipments. Finally, although it may not be immediately evident from Figure 4.4, the smallest number of items per order (2-4 items/order) has the largest squared coefficient of variation for chute-dwell time (SCV) at around 2.0, and is the only category with an SCV above 0.5. According to Hopp and Spearman (1996), as a general rule of thumb, a process with a resulting squared coefficient of variation above 1.33 can be thought of as a high-variability or out-of-control process. This is important because it means that the most frequent order sizes have the most variability in chute-dwell time.

4.2.1.3 Causal Relationships to Cycle Time Factors

In a system where flow is properly managed to account for individual item travel, dwell, and processing times, shipment cycle time will be dominated by processes which affect an entire order (such as chute-dwell time). While chute-dwell time is the primary contributing factor to shipment cycle time, it is also of value to analyze and understand any relationships which may exist among all the processes which affect the entire customer order. The pick cycle time relationship to chute-dwell times and total shipment cycle time was then explored using data from the same 400,000 orders as detailed above. Representative results from a correlation analysis between various dwell times within the order lifecycle in the Amazon Fulfillment engine are shown in Table 4.2 below.

	Pick Cycle time	*Chute Dwell*	*P-chute Dwell*	*Pick-pack*
Pick Cycle time	1			
Chute Dwell	0.59	1		
P-chute Dwell	0.05	0.06	1	
Pick-pack	0.56	0.88	0.45	1

Terminology Definitions

Pick Cycle time= Time from first to last pick of a shipment
Chute Dwell = Order accumulation time from first item arrival to last item arrival in chute
P-chute Dwell = Chute-dwell time from when order is transferred to pack chute until order is packed
Pick-pack = Total shipment cycle time from pick to pack

Correlation Value Interpretations

0.0 to 0.2 Very weak to negligible correlation
0.2 to 0.4 Weak, low correlation (not very significant)
0.4 to 0.7 Moderate correlation
0.7 to 0.9 Strong, high correlation
0.9 to 1.0 Very strong correlation

Table 4.2 Correlation of Shipment Dwell Times

While these symptoms and precise correlations will vary from system to system, three items ought to be noted.

1) Chute-dwell time is correlated strongly with total shipment cycle time. This, along with the result shown in 4.2.1.1, implies that any changes to chute dwell time should affect total cycle time with a relationship that is close to 1:1.

2) Pick cycle time is correlated with chute-dwell time, as may be expected. This means that scheduling algorithms affecting item pick times related to pick times for other items in the same order can be adjusted to reduce both chute-dwell time (chute utilization) and shipment cycle time.

3) Pick cycle time, although correlated with chute-dwell time and shipment cycle time, is not correlated strongly with either. Thus, making adjustments as proposed in 2) will not have a 1:1 impact on chute-dwell time, and may only marginally affect shipment cycle time.

4.2.2 Shipment Limiter

The shipment limiter can be an effective way to manage chute utilization and add some type of picking restriction on the scheduler. In this type of waveless system, limiting the number of possible partially picked orders is equivalent to enforcing a maximum threshold for WIP. Varying this parameter has obvious and direct consequences to the number of occupied chutes, with a possible effect on both throughput and/or cycle time. While decreasing the shipment limiter can be assumed to decrease cycle time as well, there is no enforced prerequisite on what the effect of this relationship will be. Another factor to consider is the effect on productivity. While, once again, it can be assumed that a decrease in the allowable number of partially picked orders will decrease picker productivity, the total impact will depend on the system, the scheduling algorithm in place, and the relative size of the pick labor to the rest of the system. A test on the effects of varying this inventory-restrictive parameter from a low value to a high of 2x the low value was run within the Amazon Fulfillment Engine.

4.2.2.1 Effect on Chute Utilization

While this paper has used the shipment limiter somewhat synonymously with chute utilization, it is recognized that the shipment limiter merely limits the number of partially picked orders upstream at pick rather than actual chute utilization. In order to quantify the effect of the shipment limiter on chute utilization, a five day experiment was run where the shipment limiter was varied and the resulting chute utilization was observed.

Figure 4.5 below shows regression modeling (labeled Predicted Y) on chute utilization response to shipment limiter (labeled Y) within the AFE system. The regression was valid with an $R^2 = 0.82$ and a significance value << 0.0001. While this does not provide immediate insight into the functionality of the shipment limiter, and cannot be easily generalized to other implementations of waveless picking, it does indicate the ability to roughly predict chute capacity utilization within a continuous flow system. This result can be used to decrease the uncertainty the operations team has on the chute utilization resulting from a choice in shipment limiter setting.

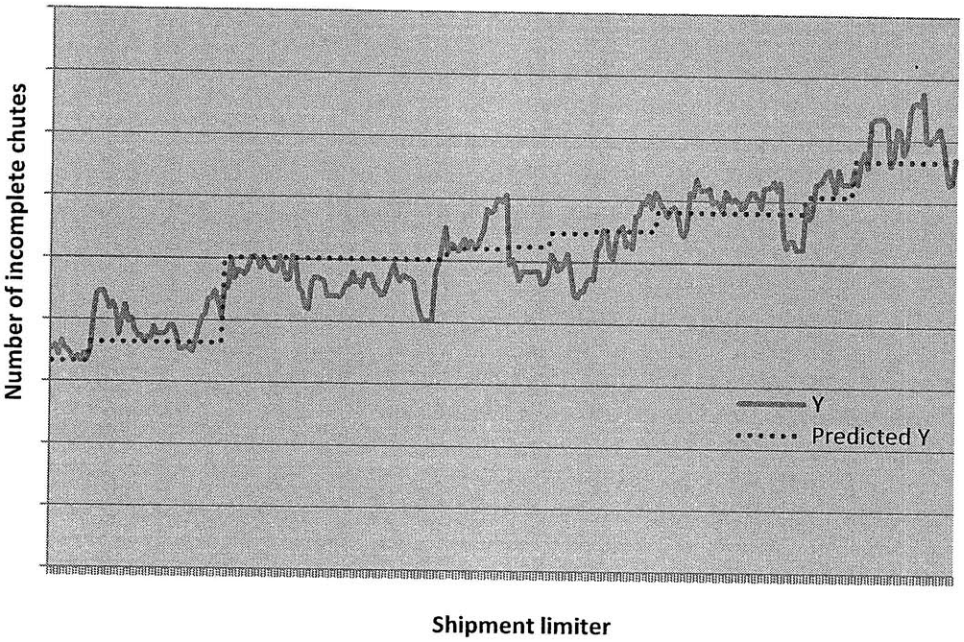

Figure 4.5 Chute Utilization Response to Shipment Limiter

4.2.2.2 Effect on Cycle Time

During the same test where chute utilization response to shipment limiter was measured, pick cycle times were recorded to determine response to the shipment limiter. As expected, the effect of decreasing shipment limiter is a decrease in the pick cycle time. Graphical results are shown in Figure 4.6, along with a linear regression to model the system behavior. The regression run had an R^2 of 0.42, and passed a significance test with a value << 0.0001.

Although managing the system by limiting the WIP at pick may appear to be a plausible way to reduce both chute utilization and shipment cycle time, a thought experiment reveals the difficulty in choosing an appropriate setting. Consider two extreme cases, first where the shipment limiter approaches one. In this case, waveless picking is reduced to be virtually identical with strict-order picking. Chute utilization will decrease, and pick cycle time will drop to almost zero, but picker productivity will also immediately plummet. Obviously this scenario is not viable to warehouse operations. On the other hand, a very large shipment limiter will result in extremely long cycle times, and will reduce the ability of a fulfillment center to meet customer promised ship date. Thus, additional indicators, including the effect on picker productivity, are needed in order to intelligently choose an optimal shipment limiter setting.

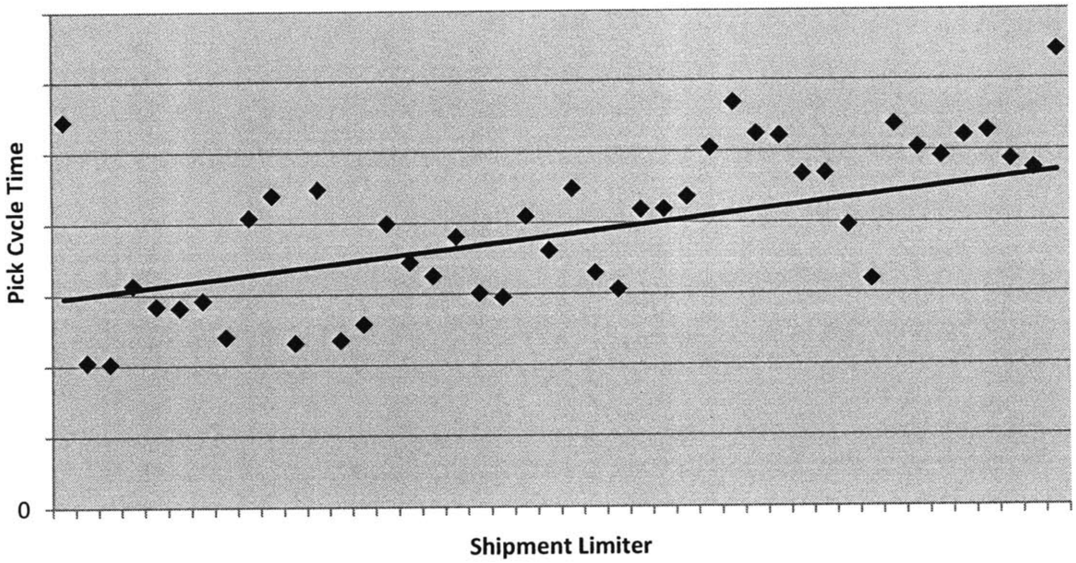

Figure 4.6 Shipment Limiter Effect on Pick Cycle Time

4.2.2.3 Effect on Pick Productivity

Picker productivity was recorded during the same test to quantify the effect of changing the allowable WIP at pick on pick density and therefore pick productivity. Within the Amazon Fulfillment Engine, pick productivity drops throughout the day (explained below), and this drop is more significant when compared to the percent drop in productivity due to decreasing the shipment limiter. Thus, the hourly data is sufficiently skewed to make all statistical tests on the model validity inconclusive, and there was not a large enough sample of daily data collected to give statistical significance to the trend of pick productivity with variation on the shipment limiter.

Since the major *critical pull time*, or time when packed orders must arrive at the dock for on-time departure, occurs at the end of the day, *shipment chasing*, or the process by which shipments are re-scheduled or force scheduled in order to meet a system constraint, increases throughout the day. This increased chasing leads to a strong hourly trend of decreasing productivity throughout the duration of the day, regardless of shipment limiter setting (refer to section 2.3.2 of this paper for additional information on how chasing may be caused for some period of time before a set ship time). See Figure 4.7 below for a depiction of how the variation in pick productivity due to chasing dominates the effect from varying the shipment limiter. Each day in this figure corresponds to one single setting of the shipment limiter, and all variation throughout the day is from chasing shipments. The data for picker productivity is expanded across each day in order to see the time-based effect of chasing, such that all data points for Day 1 have the same shipment limiter setting, and are graphed with the elapse of time. Day 2 is then a new, higher shipment limiter setting, with each data point graphed with the elapse of time, and so forth. While the regression is significant with a value of 0.005, the resulting R^2 was only 0.17, due to the extreme variations caused by chasing, which were not modeled in the regression.

These results do not capture the observed impact to adjusting the shipment limiter due to the excessive variation in hourly pick rates.

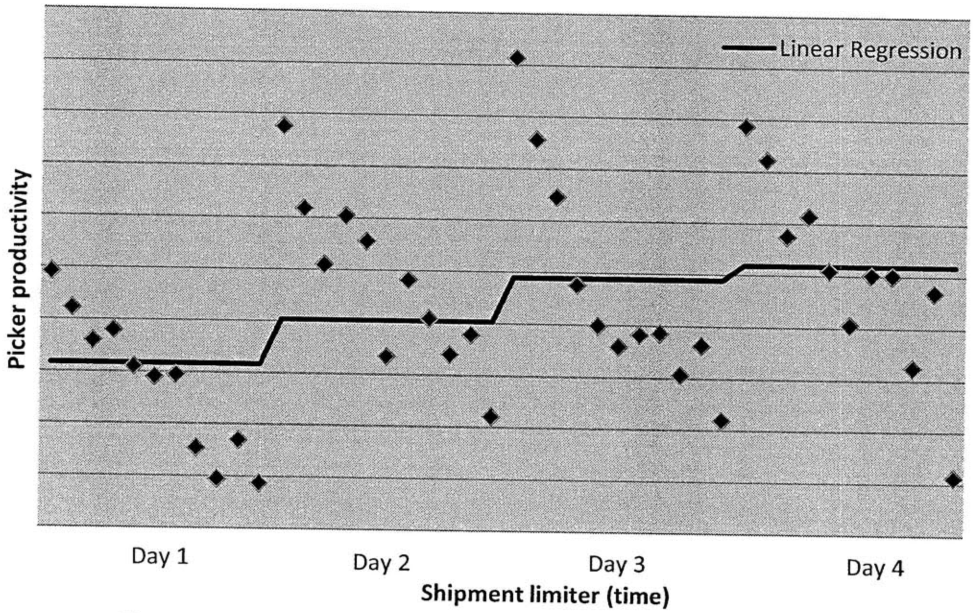

Figure 4.7 Shipment Limiter Effect on Picker Productivity

Aggregating daily pick productivity data relegates the test data to a comparison of means, and invalidates a statistical test due to the reduced sample size of only four days. However, a view of the averaged data, shown in Table 4.3 below, shows an obvious trend with the varying of the shipment limiter.

Shipment Limiter (scaled)	Pick productivity gain (% of max)	Pick cycle time gain (% of max)	Shipment cycle time gain (% of max)
70	0%	0%	0%
60	- 3%	11%	5%
50	-6%	22%	10%
40	-9%	32%	14%

Table 4.3 Shipment Limiter Net Effect on AFE

While pick productivity is a primary concern in all warehouse systems, it is clear that in a system where even 2-3% loss in productivity impacts the bottom line, a strategy for cycle time reduction involving simply restricting allowable quantity of partially picked orders is unacceptable. The only rational way to manage a system governed solely by this parameter is to decide upon an acceptable cycle time, and then set the shipment limiter as high as possible without overflowing chutes and without violating the agreed-upon cycle time. Thus, some other control must be put in place if both productivity and cycle time are critical metrics.

4.2.3 Demand Window

The demand window is an attempt to manage cycle time, since it is apparent from the previous analysis that the shipment limiter cannot manage it alone. As implemented in Amazon.com, the demand window is an average result of the underlying flow management algorithm, and specifically that its setting affects chute-dwell time in that chutes will be open for the demand window time setting *on average* over time. Although such an enforcing of pick cycle time and chute-dwell time depends on both scheduler algorithm and parameter implementation, it is reasonable to assume that results from one system are portable or at least replicable from one system to another.

4.2.3.1 Original State Analysis

A week-long test was run to observe effects of varying the demand window setting in the Amazon Fulfillment Engine. The primary management concern with enforcing strict limits on allowable pick cycle time, as with managing the allowable quantity of partially picked orders, is the effect on pick productivity. Thus pick productivity and pick cycle time were tracked while the demand window setting was varied between extreme high and low values. The results are shown below in Figure 4.8.

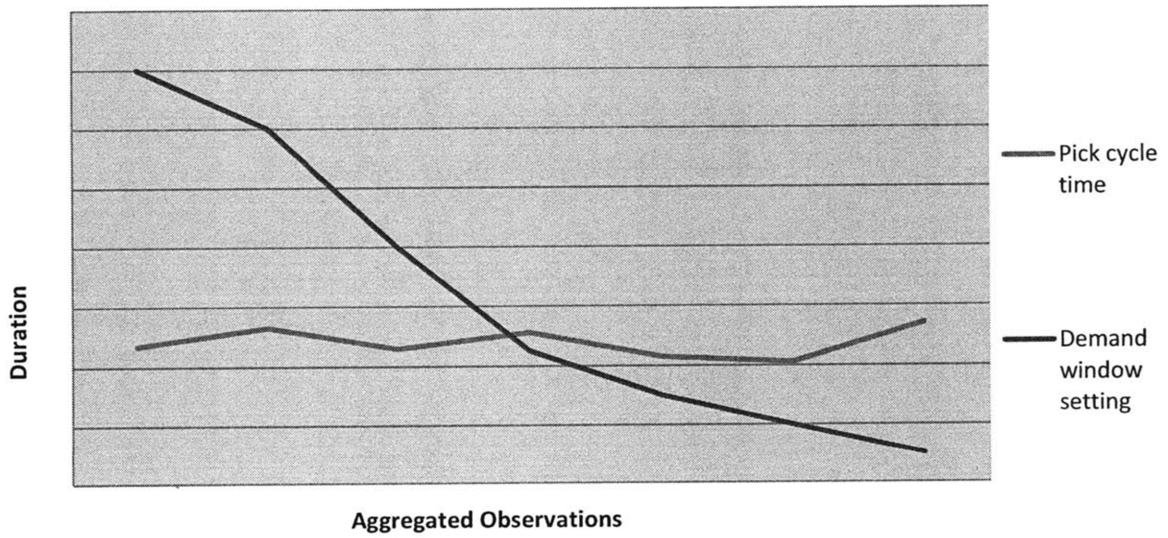

Figure 4.8 Demand Window Effect on Pick Cycle Time

Both a regression analysis and a statistical comparison of means result in accepting the null hypothesis: that there is no change in pick cycle time due to changes in the demand window. A similar comparison of pick productivities shows that pick paths were not noticeably changed from items violating the demand window setting. This inability to impact cycle time at all could be a symptom of too light an enforcement of the demand window, allowing for excessive flexibility in the scheduling of violating picks. As mentioned previously in section 4.1.1, such flexibility should be sensibly included in order to reduce chasing and the loss in productivity. However, a system that offers no ability to manage cycle time, especially when a management lever to control the same is thought to exist, reduces the ability to control system behavior with respect to customer experience.

4.2.3.2 Changes and Analysis

With the result from the original state analysis on the demand window and as part of the case study, software modifications were made to strictly enforce the demand window. In order to avoid repeating the issue of allowing for too much flexibility in scheduling picks that violate the demand window, the scheduler was adjusted to treat violation of the demand window as violation of the lateness window, or pick immediately. Aggregate results from the test are shown below in Table 4.4. Shipments chased was tracked as a sanity check to ensure that the software was actually rescheduling picks based on violation of the demand window (which did not occur in the original test).

Demand window*	Pick productivity*	Pick cycle time*	Shipments chased (%)
No limit	28	4.8	1%
8	27	5.6	5%
4	26	5.6	20%
3	25	4.6	21%
2	25	4.9	26%
1	24	4.7	30%

* Results are representative, to protect proprietary data.

Table 4.4 Adjusted Demand Window Effect on AFE

The significance of this is obvious. Despite varying the demand window, and observing the effect on picker productivity (as expected), and on the percentage of shipments chased (as expected), there is no change in pick cycle time. Obviously managing order cycle time within a warehouse can be a priority, and it has been shown that limiting WIP to match downstream resources is not a valid way to also manage cycle time.

4.2.3.3 Managing Cycle Time and WIP

At first the result obtained in 4.2.3.2 can seem perplexing, that forcing picks based on pick cycle time should have no effect on pick cycle time. However, a thought experiment reveals the possibility of over-constraining the picking scheduler. One of the physical limitations to automated sorters is the chute capacity to accumulate orders. Therefore, a maximum threshold is required to limit the chute utilization to a quantity that the system can handle. Such a constraint immediately causes a reduction or potential reduction in picker productivity, so a natural response may be to maximize the chute utilization within the bounds of the system capacity. This strategy corresponds to using the shipment limiter as a target value, requiring a certain number of orders to be partially picked to maximize pick labor productivity rather than limiting to a maximum value. Referring back to the queuing formula $\lambda=L/W$, constraining WIP (orders allowed into the system) is identical to constraining L. Noticing the decrease in productivity resulting from additional enforced chasing, in order for *W* to remain constant, which in this case scenario occurred, productivity must decrease at the same rate that demand window violating picks are rescheduled to be picked earlier.

For example, consider a shipment limiter that is set at 500. If orders 1-20 violate the demand window, then they must immediately be scheduled for picking. While picks for these ten orders are being picked, two things can be thought of as occurring:

1) The other 480 orders are being slightly delayed from being picked, thus increasing the probability of adding to the pool of orders that violate the demand window

2) New orders are being immediately scheduled for picking to meet the shipment limiter requirement of 500 partially picked orders

The occurrence of 1) means that while pickers are in the process of "catching up" with the required pick cycle time, additional orders are falling behind, thus keeping productivity down. The occurrence of 2) implies that, while productivity is falling and new orders 501-520 (to replace orders 1-20) begin to be picked, the time required before orders 501-520 can be completely picked increases above what it otherwise would have been. Now if the same scenario is compared with orders 1-40 violating the demand window (for example, due to a relatively shorter demand window setting), productivity will drop further, and the time will be that much longer than it otherwise might have been to completely pick the new orders falling into the system. Thus, *by design*, and by the fact that meeting the shipment limiter is a primary constraint (if items in many orders are dwelling beyond the demand window, but only 490 shipments are currently partially picked, then the scheduler will give priority to scheduling the opening of 10 more shipments above completing demand window-violating picks), productivity will drop at a rate that leaves cycle time constant. Thus cycle time cannot be impacted by adjusting pick scheduling as long as the system remains WIP-controlled.

4.3 Tote Prioritization

One of the key management issues in managing cycle time, and specifically chute-dwell time, relates back to the relationship between chute-dwell time and pick cycle time discussed in 4.2.1.3. Revisiting the issue, essentially two methods exist for decreasing chute-dwell time: reducing pick cycle time, and adjusting downstream processing to reduce the variability in chute-dwell time that is not accounted for in the moderate correlation between chute-dwell time and pick cycle time. One such possibility exists in prioritizing pick totes sent to induct as they arrive to the sortation system.

4.3.1 Basis for Optimization

The current AFE system transports totes from pick to induct in a first-in-first-out (FIFO) manner. A circulating buffer exists just prior to the tote arrival shown in Figure 2.4, but is currently used exclusively for overflow in the event that the induct stations are backed up, potentially blocking the conveyor from delivering additional totes from pick. The idea behind tote prioritization is that recirculation buffers can be used to prioritize and selectively send items downstream for further processing. In the case of full tote prioritization, the goal is to find a way to send totes with a greater quantity of closers than other totes arriving at the recirculation buffer, thus providing a way for a greater number of closers to arrive at induct and sortation more quickly than they otherwise would. This in turn can reduce chute utilization by decreasing chute-dwell time, which can increase throughput by offering additional, usable capacity. The relevant performance measure to determine the effectiveness of this scheme is therefore chute-dwell time.

The method for prioritizing totes can be thought of as "scoring" totes based on the number and type of their contents, then comparing scores among recirculating totes to determine priority. The optimization model tested here does precisely that, with an objective function of minimizing chute-dwell time, as shown in Equation 4.2, and the following decision variables:

1) Opener score where the closer has already arrived at the sorter
2) Opener score where the closer has not yet arrived at the sorter
3) Single score
4) Middler score
5) Closer score
6) Number of totes allowed in recirculation

These scores can then be used to determine tote score by finding the average score of the tote as seen below in Equation 4.3. Totes can then be released to induct with all higher scoring totes sent first.

$$min \sum_{j=1}^{m} t_j$$

Where: t_j = chute-dwell time of order j
m = total number of orders within a simulation run

Equation 4.2 Model Objective Function

$$\frac{\sum_{i=1}^{n} y_i}{n}$$

Where: n = number of items in a particular tote
y_i = score for item i in a particular tote

Equation 4.3 Tote Scoring Calculation for Prioritization

4.3.2 Modeling

The item arrival and sortation system in the Amazon Fulfillment Engine was modeled with the ProModel Solutions 7.5 discrete-event simulation modeling package. The simplifications and assumptions made in the discrete-event model are:

- Tote arrivals are assumed to occur at even intervals equal to the empirically observed mean inter-arrival time

- Actual tote density (number of items per tote) was tracked over a one-week period, and model tote densities are probabilistically assigned according to that distribution

- Tote contents (number of singles, openers, middlers, and closers) are determined randomly in the following way
    - Average number of items per order for multi-unit orders was observed over a period of days
    - Openers, middlers, and closers are randomly assigned based on the ratio 1:(x-2):1, where x is the average number of items per order observed
    - Single items are used as fillers to match model tote density with actual tote density, and are assigned at the same frequency as empirically observed

- A target value is set for number of orders which have partially arrived at the system. The driver for this is to match the effect of managing using the shipment limiter. Additional variation is allowed to match actual system response, which has higher variability in chute utilization than in number of partially picked orders due to the variability added during the pick and travel process.

- Processing rates for induct labor and rebin labor are assumed constant at the appropriate mean observed labor rate

- All travel times are assumed constant, with each leg of travel approximately matched to the average travel time associated with the various conveyor locations.

- For model simplification, individual item queuing at final sort was not modeled. In the AFE system, these queues are less than 2% of total cycle time, and are considered insignificant. Modeled items are immediately assigned a chute upon arrival to final sort.

Model product flow is similar to the graphic in Figure 2.4, where totes arrive based on the logic described above, then are split into individual items at induct, and items are finally sorted into individual chutes. The model includes a tote recirculation lane prior to tote arrival to sortation, which mirrors actual system implementation. This recirculation lane was the basis for modeling, to determine the effect of using the recirculation lane as a way to prioritize tote order before sending product to induct. Base-case modeling follows current system performance of only using recirculation if totes are stacked all the way back to tote arrival, thus blocking additional totes from entering the system.

4.3.3 Model Validation

The discrete even simulation model was validated by tracking chute-dwell time over a period of five days. During this five day period, the shipment limiter was varied among four values: a low extremity, medium low, medium high, and a high extremity. Model output of chute-dwell times using the same shipment limiter settings was then compared to actual hourly average chute-dwell times. This method for validating the model was chosen for the following reasons:

1) Imitation of reality – The current AFE system is managed primarily by adjusting the shipment limiter. Thus, comparing the model response to variations in the same management lever used in the actual system management is appropriate.

2) Breadth of validation – By testing model output at both extremities of the shipment limiter setting as well as two intermediate values, the entire spectrum of possible responses are covered within the tested range.

3) Minimal invasiveness – Due to the labor implications of adjusting the shipment limiter, limiting the shipment limiter to four values reduced productivity effects and limited the duration of the data-gathering stage to quantify actual system response.

For purposes of model validation, all orders which spanned a shift change, lunch, or worker break were backed out to give the chute-dwell time *only while the system was actually in operation.* This ensures that actual data is not artificially inflated, causing the false appearance of a discrepancy between the model output and the observed data. Figure 4.9 below shows the actual average chute-dwell times together with the model output chute-dwell times.

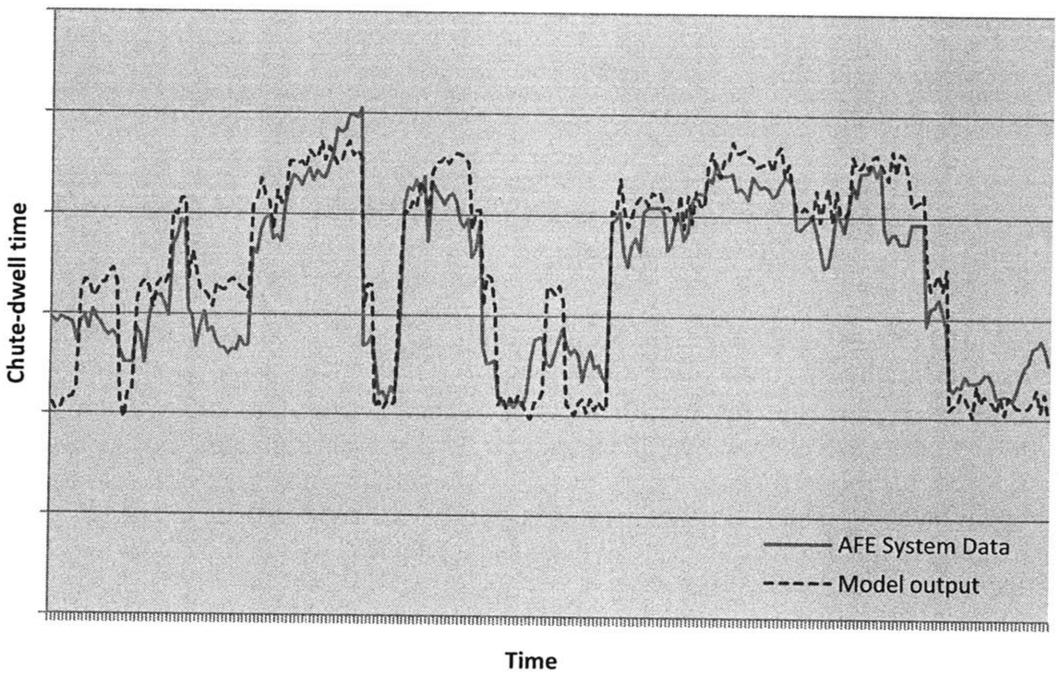

Figure 4.9 Simulation Model Output vs. Actual System Response

A percentage error calculation for each pair of data points in Figure 4.9 above, using the relationship indicated in Equation 4.3, reveals that the model tracks reality to within 10.5% on average. This value indicates that the model output is reasonably consistent with the actual data, especially when compared to the internal system variability of chute-dwell time of 4.2% (for a given shipment limiter setting). The internal system variability was also calculated using Equation 4.3, with each x_i occurring during the same shipment limiter setting, and using the average of all x_i's as$\hat{x}_i$.

$$Average\ percent\ error = \frac{\sum_{i=1}^{n} \frac{|x_i - \hat{x}_i|}{x_i} * 100}{n}$$

Where x_i = actual chute-dwell time for observation i
$\hat{x}_i$ = simulated chute-dwell time for observation i
n = total number of observations

Equation 4.4 Simulation Model Error Calculation

With the visual similarities between modeled chute-dwell time and actual chute-dwell time, along with the percent error of 10%, and the realization that this is only slightly over 2x the inherent system variability, the model can be accepted as valid for the case in study. Methods for model optimization can then be considered, as the decision variables described in Section 4.3.1 are varied in order to achieve the objective function and minimize chute-dwell time.

4.3.4 Model Optimization

A simple method to differentiate between possibly higher and lower priority totes was chosen: a scoring of different "types" of items. Items were categorized as singles, middlers, closers, or openers. Openers were then further divided into two categories, openers to orders for which the closer had not yet arrived to the AFE system and openers for which the closer had arrived to the AFE system. These categories were chosen for two reasons:

1) Historical Basis – They resemble categories used in other sortation systems as a basis for prioritization (Gallien, July 2009)

2) Simplicity – They are simple to implement since all items and totes can be scored and updated at the sorter, with no need to communicate back to pick software. (In contrast to other prioritizations which assign additional categories such as openers to orders for which the closer has not been picked yet)

The ProModel internal simulation and optimization runner was employed by iteratively:

1) Assigning values to each of the six decision variables

2) Running a simulation with the chosen values

3) Comparing decision variable values and average simulation run chute-dwell time with previous runs

4) Intelligently choosing new decision variable values in an attempt to converge to an optimal solution

The constraints placed on decision variables within the ProModel optimization algorithm were chosen to bound the problem and reduce the needed run-time for convergence to an optimal solution. Decision variables that did not have obvious priority were given a full range from -100 to 100. Item types that should obviously be of higher priority (e.g. closers) were limited to the top half of the same range, or 0 to 100, and types that were obviously of lower priority were limited to the low half of the full range. The tote recirculation decision variable was limited to physically realizable values, or 0 for a minimum, and the recirculation lane capacity of 80 for a maximum. Table 4.5 below details the full list of constraints placed on the ProModel optimization software.

Decision variable	Min Value	Max Value
Opener (closer not arrived at sorter)	-100	0
Opener (closer arrived at sorter)	-100	100
Middler	-100	100
Closer	0	100
Single	-100	100
# Totes in recirculation	0	80

Table 4.5 Constraints on Decision Variables

A tracing of the optimization runs, with decision variable settings for each successive run chosen automatically by the internal ProModel optimization software, is shown in Figure 4.10 below.

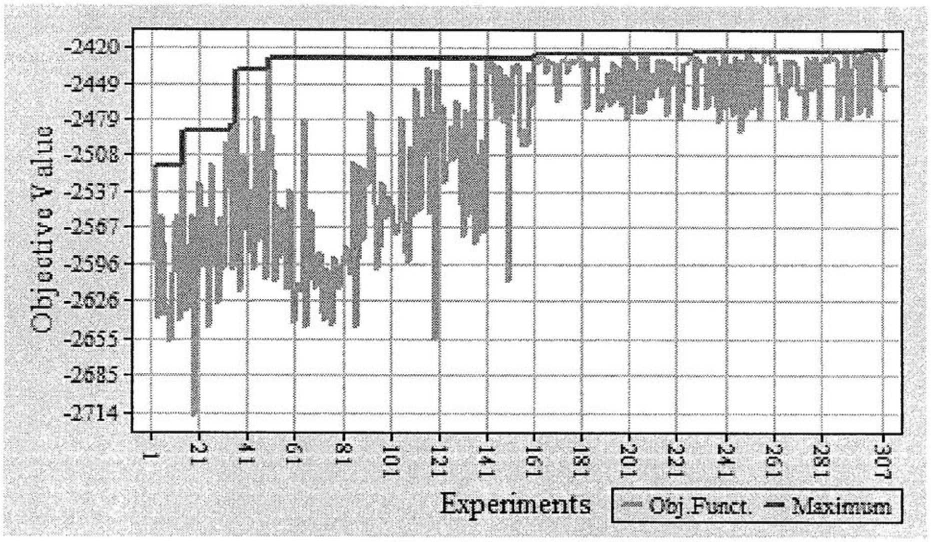

Figure 4.10 Tote Prioritization Optimization Run Plot

Near the end of the optimization runs (approx. runs 161-307), the objective value did not visibly improve, and in fact seemed to oscillate rather than converge further towards optimality. Thus, a more accurate description of the outcome of the simulation runs can be represented by a 95% confidence interval around the optimal settings found from simulation runs. This was obtained by finding all simulation runs with identical (to within 1%) 95% confidence intervals as the 95% confidence interval for the optimal run. The min and max decision variable settings for each variable within this subset of simulation runs, was then recorded as the 95% confidence interval for each parameter. This information is recorded in Table 4.6 below, and gives a feeling for the sensitivity of the optimal settings found through the optimization process.

Parameter	Optimum Score	Low 95% CI	High 95% CI
Singles	-6	-14	2
Openers (closer not arrived at sorter)	-44	-52	-36
Openers (closer arrived at sorter)	14	-9	37
Middler	1	-1	3
Closer	7	6	8
# Totes in recirculation	49	48	50

Table 4.6 Sensitivity of Tote Prioritization Optimization Results

The importance of this sensitivity analysis is two-fold. First, a range of plausible decision variable values has been calculated, with 95% probability that the actual optimal setting is contained within that range. This significantly narrows down the choice for possible settings when actually implementing recirculation. Second, as indicated by Figure 4.10, even if the optimal settings are not precisely known, being slightly off will result in an almost identical response to chute-dwell time as the optimal settings. The optimization results and implications are discussed in more detail in Chapter 6.

4.4 Summary

One method to achieve throughput increases in fulfillment warehousing is by reducing shipment cycle time while holding WIP constant. Chute-dwell time typically accounts for the vast majority of cycle time, and is a primary candidate for focus in cycle time reductions, both for throughput increases and to reduce current capacity utilization. Chute-dwell times can essentially be reduced in two ways: 1) decrease pick cycle time through scheduling improvements, or 2) reduce variation in processing between pick and order accumulation. Due to the only moderate correlation between chute-dwell time and pick cycle times, each is a viable possibility for realizing gains.

Waveless picking systems can be controlled by managing the number of allowable partially picked orders (shipment limiter), and/or by enforcing limits on the allowable pick cycle time (demand window). Each method has implications on picker productivity, chute-dwell time, and capacity utilization, but which parameters are directly and indirectly affected differ between them. A case study analysis of the Amazon Fulfillment Engine reveals the system limitations of attempting to manage both cycle time and chute utilization. Limiting the allowable quantity of partially picked orders while trying to maximize picker productivity leads to an inability to have any direct effect on pick cycle time. Conversely, limiting the pick cycle time leaves questions about how to avoid overflowing system capacity and how to realize picker productivity; a key issue for many adopters of waveless picking systems. Planned recirculation can be another effective method for reducing chute-dwell time and limiting capacity utilization for increased throughput, though there is no effect on shipment cycle time for improved customer experience.

Chapter 5: A Comparison of Wave and Waveless Picking

Wave picking has been a standard picking method for many years, and only in more recent years has waveless picking become a viable option. Despite the advantages that waveless picking proponents claim are immediately observable after adoption of continuous flow picking, many companies have yet to adopt such a picking strategy for their distribution warehouses. This chapter provides a side-by-side comparison of wave and waveless picking, using both previous literature and studies along with the learnings obtained from the case study considered in Chapter 4 of this paper.

5.1 Perceived Barriers to Waveless Picking

Aside from the possible capital investment required to implement waveless picking, for items such as two-way RF scanners, software development, automated sort systems, etc, additional concerns may deter corporations from adopting waveless methods. Some common system concerns that may be seen as more possible in waveless systems are congestion, gridlock, and complexity (Gallien August, 2009 and Peterson 2000). Additional barriers noted by the author of this paper are customer experience and control. Each is discussed briefly below.

5.1.1 Congestion and Gridlock

Order release strategy is known to have an effect on congestion, though that relationship varies among systems. Obviously wave systems can have similar congestion issues as a waveless system, through poor scheduling that does not allow orders in a wave to complete, or through overly-complex overlapping of waves in an attempt to maximize the sorter and system capacity. Waveless picking can be seen as more risky in terms of congestion simply because it is designed to allow more flexibility by the scheduling software, with no hard-line enforcement of closing orders out. Thus, a waveless picking system may appear to be more easily susceptible to congestion occurrences for the same number of orders and items as a corresponding wave-based algorithm.

The presence of congestion itself is not so much a concern, except in that increased congestion causes longer cycle times and an increased probability of gridlock. When gridlock occurs it is a systemic issue, and can be very costly and time-consuming to undo. Sam Flanders of 2wmc.com, quoted in Bradley (2007), compares gridlock in a waveless system to solitaire. He says, “If all the slots in the game are full, the game is over, and you lose. If you have 10,000 SKUs and 1,200 drop points, you can have a lot of SKUs on the sorter with no place to drop into. If you want to work with continuous flow, you have to be cognizant of this.”

Gallien and Weber (August 2009) performed a simulation-based analysis on both waveless and wave-based systems to observe the incidence and effect of congestion and corresponding probability of gridlock. They refuted the hypothesis that gridlock is always more probable in a waveless system, noting that the waveless policies studied can result in lower gridlock probability than the best-performing, sensible implementation of a wave-based policy.

They noted that the major drawback to this is that such strong performing waveless systems may not be sensible to implement, as the complexity of developing such solutions can be significant, and companies may lack both the expertise and the resources to warrant a switch to a high-performing waveless system.

5.1.2 Complexity

Arturo Hinojosa, one of waveless picking's more vocal enthusiasts, notes why the implementation of a waveless system is non-trivial. Issues that must be considered and analyzed in great detail during the design phase of such a system include labor balancing, pick zone synchronization, real-time replenishment, and labeling requirements (Hinojosa 2006). Management of the continuous flow of waveless systems can be more difficult, as constant and real-time updating of labor, scheduling, and constant management of queue sizes may be necessary.

Complexities may also exist simply by design in the picking system itself. Figure 5.1 illustrates the types of objectives manufacturing organizations typically face, and how complexity may arise through opposing goals. The continuous balancing of pick productivity with throughput and cycle times is an example of the classic struggle between utilization and inventory, as exhibited in the figure below. In contrast, a wave-based system is well understood, and does not require the same amount of daily considerations on how to deal with management complexities.

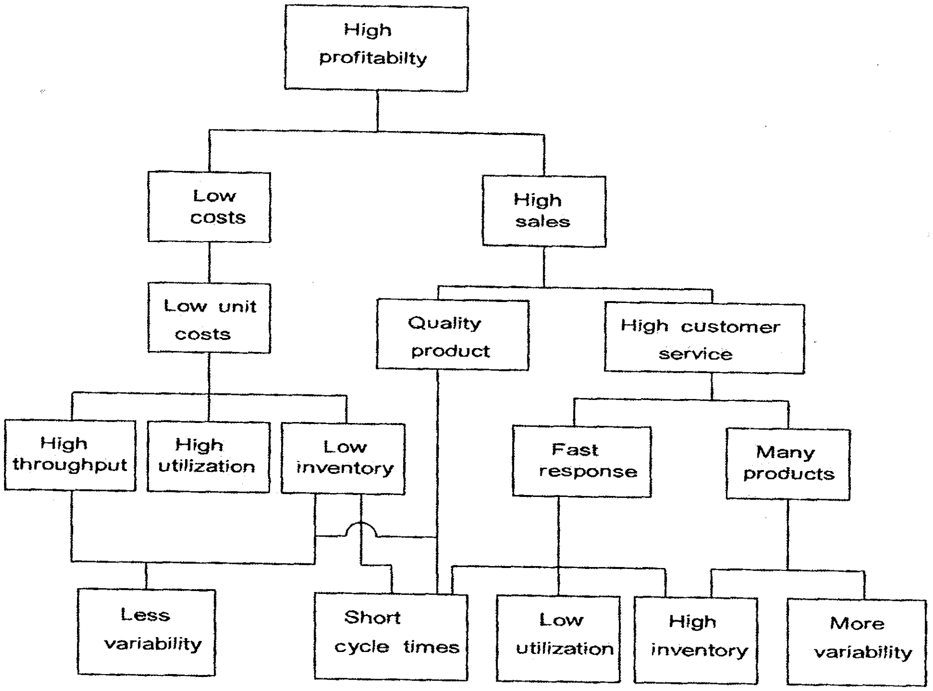

Figure 5.1 Hierarchical Objectives in a Manufacturing Organization[4]

[4] Figure copied from Figure 6.3 in Simchi-Levi (2000)

5.1.3 Control and Customer Experience

It is self-evident that it is desirable to have more ability to control and predict a process than otherwise. As described in 4.2.3.3, one of the effects of the design of a waveless picking system, or at least the implementation adopted by Amazon.com in the Amazon Fulfillment Engine, is that system control over shipment cycle time and the precise predictability of chute utilization is decreased. This reduction in control, as exemplified in the case study in this paper, can limit the warehouses ability to meet customer experience. For example, consider a warehouse where:

- Orders are allowed to drop in to the system at 6pm with a promised arrival date of the next day
- Final outbound shipments have to be on the docks at 9pm to make next-day arrival
- There is a one-hour processing time that excludes chute-dwell time (i.e. the expected processing time from pick to ship of a closer is one hour)

With more control over processing (such as would be the case in a wave-based system), the warehouse has the flexibility to choose to allow customer orders to drop in to the system up to an even later time, and simply create smaller overlapping waves. In a WIP-controlled waveless system, with no direct control over cycle time, orders allowed to drop in later have no guarantees that they will be scheduled appropriately to be picked in time to meet the ship date, especially if there are a significant number of them.

One item worthy of noting is that customer experience has been stated to be one of the benefits of waveless picking. Hinojosa (2006) states that waveless picking offers enhanced customer service through better shipping deadline management, asserting that the flexibility offered by waveless picking to control real-time which orders are picked, can be used to prioritize orders in a fashion that would be equivalent to re-prioritizing orders mid-wave in a wave-based system. While this is not in dispute, a few observations must be made about such a situation:

1) Prioritizing key shipments that exceed the lateness window is fundamentally different from achieving cycle time control. While both can affect customer experience, one involves real-time flexibility in scheduling, while the other involves flexibility in control.

2) The waveless picking system must be designed with sufficient flexibility to allow switching between priorities of productivity and cycle-time control. Such a system would, as a result, have increased management and implementation complexities instead.

3) Quantity differentials must exist between the number of priority and non-priority orders in order to avoid overwhelming the scheduler in its attempt to not fall behind. In a wave-based system, if no quantity differential exists, smaller waves with sufficient labor can still achieve low average pick cycle times, while a WIP-controlled warehouse will be at the mercy of the underlying algorithm.

5.2 Perceived Enablers to Waveless Picking

Waveless picking has been seen by many as "the wave of the future" (Bradley 2007). Although many possible gains are touted by waveless picking advocates over wave-based systems, two are most consistently named as the most prominent and immediately realizable advantages: productivity and throughput. Some additional side-effects of waveless picking which can benefit an organization are lower investments needed for new projects, better handling of last minute orders, and flexibility allowing for order prioritization to maximize customer experience (Hinojosa 2006). While these factors are not analyzed below, they also play a part in the decision-making process for warehouses considering adopting this new technology.

5.2.1 Productivity

Productivity gains are the most prominent benefits that potential waveless picking adopters stand to gain from an overhaul of their scheduling systems. Pick productivity, which typically accounts for from 50-60% of the operations costs in a fulfillment warehouse, can be immediately improved from the increased pick density offered by allowing all picks to be scheduled from the same pool simultaneously. This paper has indicated how chasing the productivity gains too much can lead to additional limitations, but the potential gains in this area cannot be ignored. Bradley (2007) quotes Fortna Inc. (a supply chain consultancy and systems integrator) on potential gains of up to 20% in picking productivity achieved just from adopting waveless picking.

Not only can real-time scheduling increase pick density and thus help increase picker productivity, but it also reduces the amount of straggling time of pickers, which further increases pick productivity. *Straggling time* is the time spent by pickers who have already completed their picks, as they wait for the straggling pickers to finish so that a new pick lists can be created. Thus, wave-based picking causes straggling time between waves simply because pickers do not all complete their pick lists simultaneously. Those pickers that finish earlier must wait for the so-called stragglers to complete their picks before all pickers can move on to the next wave. Significant effort has gone in to reducing this straggler effect, through overlapping waves and other process innovations. However, despite all this, the reduction in productivity due to transitions between waves does not go away. Figure 5.2 below gives an empirical example of how pick productivity decreases at the end of waves, and how even when using overlapping waves, there are still periods of reduced productivity during wave transitions. This figure displays histograms of the number of picks completed for overlapping waves during successive 5-minute intervals. The end of each wave has significantly fewer completed picks than in the middle of the wave when there is no loss in productivity due to wave "changeover".

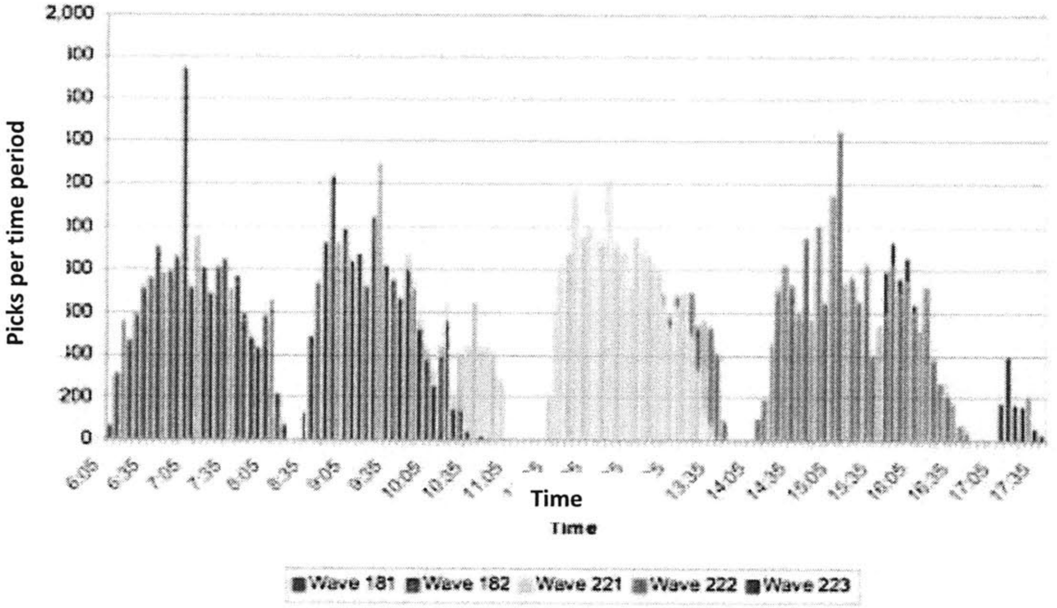

Figure 5.2 Productivity Reduction During Wave Transitions[5]

5.2.2 Throughput

Throughput in wave-based picking is severely limited by the time-based under-utilization of system capacity. As mentioned in 5.2.1 of this paper, wave picking involves separation in time of various waves. This loss in productivity, which cannot be completely recovered, flows downstream and results in lost throughput as well. The net effect of these transition periods can reduce sorter utilization by 60% or more (Perry 2007). Attempts to mitigate the effects of throughput reduction due to wave transitions include adding queues to ensure continuous availability of work, and re-assigning unproductive labor to other locations within the warehouse. However, attempts to minimize the effect of wave transitions add double-handling and other inefficiencies of their own. Thus, low sorter utilization (and therefore throughput) is usually just accepted as "a fact of life" (Perry 2007). In Gallien and Weber (August 2009), a simulation analysis was used to compare wave-based and waveless release policies and their effect on throughput. In the scenarios considered, throughput could potentially be higher in waveless policies than otherwise. Once again (as seen in Section 5.1.1 of this paper), the major hurdle for achieving the throughput gains over wave-based policies is observed to be complexity and necessary resources to implement a sensible and optimal waveless system.

While this paper has explored some of the limiting factors on throughput in waveless systems, total throughput can still be higher when compared to wave-based systems. While chute capacity, the lateness window, the demand window, and the shipment limiter may impose throughput limitations below the capacity of the sorter, at least a subset of them is necessary to control and manage flow. This tradeoff, while it exists, still offers gains throughput gains of up

[5] Figure extracted from Chart 1 in Hinojosa (2006)

to 35%, and is not so much a tradeoff compared to wave-based picking, but an internal tradeoff within a waveless system between throughput, productivity, and complexity (Hinojosa 2006).

5.3 Summary

Productivity and throughput improvements, the historically perceived enablers of waveless picking, are strong motivators for adoption of waveless picking. Gains of up to 20% and 35% respectively, can be achieved in a waveless picking system over a wave-based system, and in no cases considered, in literature treating the subject, was a wave-based system able to compete with a waveless system which has been designed and optimized based on these two measures.

The perceived barriers to waveless adoption include congestion, gridlock, and complexity. Complexity is a significant and valid concern, and the case study of the Amazon Fulfillment Engine performed in this paper adds additional considerations on the control and management of a waveless system which have not previously been considered. Congestion and gridlock, however, can turn out to be an enabler to waveless picking. Probability of gridlock can actually be decreased through adoption of a waveless picking system, thus further increasing productivity and throughput by avoiding the fire-fighting necessary to reverse the effects of gridlock. Finally, in the Amazon Fulfillment System, it was discovered that the ability to meet customer experience can be an added barrier to waveless picking adoption. Reduced control over system performance in picking long-dwelling shipments can lead to losses in customer experience which can easily be avoided in a wave-based picking system.

The choice of a waveless or wave-based system depends on the drivers for implementing the warehouse. Wave-based picking has advantages if there is a constraint on the investment available to get an operation rolling. Also, if complexity is a concern, wave-based picking can be a simple solution. However, every warehouse has some consideration for productivity and throughput, and these are both major drivers for choosing a waveless system over a wave-based system. As long as complexity and investment can be appropriately managed, serious consideration should be given to adopting waveless picking to gain the benefits offered there.

Chapter 6: Results and Conclusions

The results and conclusions reached as an outcome of the research and experiments supporting this paper have mostly been discussed in Chapters 4 and 5. In this chapter, these results will be reviewed, summarized, and expanded upon in a single forum. Future research opportunities as well as case-specific results are also discussed.

6.1 Over-constrained Pick Scheduling

In an attempt to get around the complexities and lack of control potentially offered by waveless picking, the implementation of the underlying scheduling algorithm might over-constrain a waveless picking system, resulting in the façade of control where none actually exists. An example of this is the Amazon Fulfillment Engine, where the system is primarily WIP-controlled, with the shipment limiter as the highest priority control parameter. While this offers the control over chute utilization, it gives no specific control over average or maximum allowable pick cycle time or chute-dwell time. Thus, any attempts to enforce boundaries on cycle time result in an over-constrained problem where one of the control parameters must be ignored by the scheduling software.

A warehouse with limited chute capacity logically needs some type of control to ensure that chute overflow does not occur. Since this limits the pool of pickable items, this also decreases picker productivity. Thus it could also be logical to maximize the shipment limiter control within the bounds of the chute capacity. While this leads to higher productivity, this also *by design* leads to higher cycle times due to the higher inventory levels. Thus, increased productivity by this standard *also* means higher chute utilization and longer chute-dwell times, with the possible cascading effect of reduced throughput. The graphic below in Figure 6.1 displays this simple relationship discovered in such a system. This graph was obtained by varying the shipment limiter over a small range of intermediate values for a period of several weeks, and then running a linear regression on productivity and cycle time. Both regressions were significant and with an R^2 above 0.7 in either case.

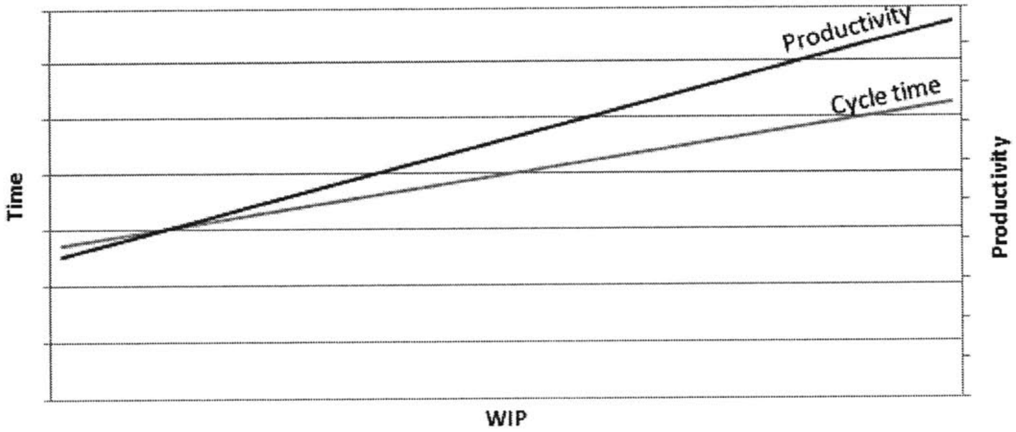

Figure 6.1 Productivity and Cycle Time Trend with WIP

Managing instead by cycle time offers more specific control over cycle time and thus customer experience, but also leads to uncertainty in the ability to ensure that there is sufficient chute capacity to meet upstream picking requirements. Thus, attempting to add cycle time contraints to to an existing WIP-controlled system for increased throughput will have no system effect. Instead, a choice exists on whether to switch control paramaters rather than add additional control parameters. This overarching lack of system control could possibly be an inherent trait within a waveless picking system, and certainly existed in the specific implementation studied in conjunction with this paper.

6.2 Tote Prioritization

Prioritizing incoming totes to the sortation area of a warehouse provides a method to decrease chute utilization and chute-dwell time if that is a limiting constraint. While this does not decrease (and may possible increase) total shipment cycle time, it can be used as an improvement mechanism where total turn-around time is not the primary driving concern. The possible realizable improvements are

1) Improved productivity – The decreased chute utilization for a given shipment limiter could be used to increase the shipment limiter and chute utilization, resulting in higher picker productivity.

2) Increased throughput – In some systems, high chute utilization can cause congestion and result in under-utilizing a sorter (due to recirculation utilization) that may otherwise be able to be fully utilized. Decreasing chute utilization from tote prioritization can mitigate this effect, and increase total throuhput.

The simulation case study performed on tote prioritizaiton within the Amazon Fulfillment Engine resulted in a tote prioritization scheme based on simple metrics, which gives an 11% reduction in chute utilization. Figure 6.2 below gives an example simulation run result set on chute utilzation for the current state and the optimal state for prioritized totes.

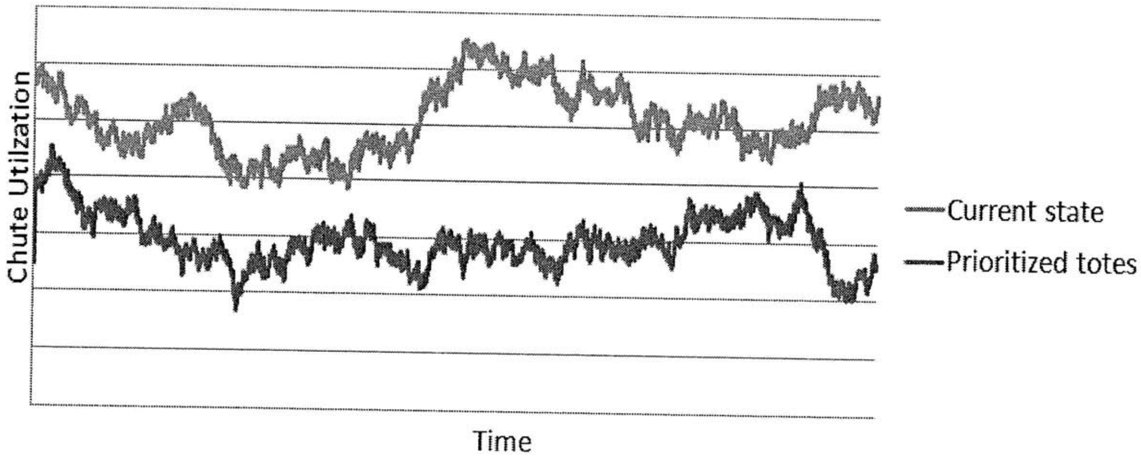

Figure 6.2 Chute Utilization Improvement with Prioritized Totes

While quantification of the actual effect on productivity and throughput will be system dependent, it is obvious that the shipment limiter can be increased to allow an 11% increase in chute utilization for picker productivity improvement with similar downstream system performance as in the current system. Throughput increases could be similarly calculated on a system-by-system basis.

6.3 Increased Throughput Realization

One of the key discoveries coming out of the research backing this paper is the applicability of the improvement methods previously discussed. For example, increasing chute capacity (or decreasing chute utilization) has a system effect only if chute capacity is currently limiting system performance, either by limiting sorter utilization or by causing increased probability of gridlock. Decreasing shipment cycle time also only has an effect on throughput if each workstation is sufficiently under-utilized that the cycle time reduction can translate into increased throughput rates instead of simply a change in the volume of work in process. To ensure that the improvement mechanism is not mis-matched with the bottleneck process, a simple analysis of possible offending factors must be performed in the following order:

1) Mechanical capacity analysis – Physical, mechanical limitations must first be addressed to ensure that the system can process at the desired throughput rate

2) Operator capacity analysis – Once mechanical limitations are overcome, operator capacities must be analyzed and adjusted to be above the sorter capacity to avoid shifting or dual bottlenecks

3) Chute availability and utilization analysis – Chute utilization can then be analyzed to ensure that congestion is not the cause for limiting throughput. *This is often the first assumed bottleneck, since chute utilization can be maximized by increasing the shipment limiter, even if this is not causing the sorter to be under-utilized and upstream processing is already at maximum capacity.*

4) Sorter capacity and utilization analysis – Sorter capacity, which represents the most costly and difficult factor to adjust, can finally be analyzed for potential improvement once all of the above factors have been analyzed. *If chutes are artificially over-utilized, it could conceivably and wrongly be assumed by an operations team that the sorter is able to run at full capacity, and is limited only by chute utilization. This could lead to efforts in increasing sorter capacity, when in fact perhaps the system is limited by upstream processing capacity instead.* This can be verfied by checking system throughput not based on congestion on the sorter, but as compared to the theoretical throughput capacity described by Equation 2.1 of this paper.

The case-study in the Amazon Fulfillment Engine was prey to skipping to steps three and four above, without fully completing steps one and two. Thus, while the discoveries on limited system control and tote prioritization can be used in future improvements to increase system throughput, immediate throughput increases were achieved through capacity analysis and mechanical and process improvements to match processing capacity at or above sorter capacity.

Both wave-based and waveless systems can benefit by following the analysis and improvement steps in the proper order.

6.4 Future Research Opportunities

The following future research opportunities exist to build upon the learnings from this paper, and to more fully characterize waveless picking system optimization methods and the improvements gained over wave-based systems by adopting the newer technology:

- The case-study AFE analysis of chute-dwell time correlation to pick cycle time resulted in a mediocre correlation value of 0.56. This paper researched only one possible explanation for the added variability in chute-dwell times. Additional research on the specific drivers to this additional variability can provide insight into process improvements to decrease chute utilization and increase its dependency on pick cycle time.

- While this paper notes the difficulties in managing a WIP-controlled system, and suggests the challenges to a cycle-time controlled system, additional research can give insights into the extent of these challenges.

- Optimization methods for scheduling picks within a cycle-time controlled system were not explored, offering another opportunity for future development.

- The tote prioritization modelling included several limitations which can be researched to better quantify the effects of various prioritization methods. Some of these limitations include:
    - Other prioritization methods were not considered outside of a simple scoring mechanism for five classifications of items
    - The target number of totes in recirculation, while used as an optimization variable, was not allowed to vary within a run. Additonal research on how to dynamically prioritize totes not just relative to other totes in recirculation, but relative to a "very important" threshold can also be considered.

- A specific study of how to truly manage both WIP and cycle time within a waveless system (either time-varying or through "loose" enforcements of both WIP and cycle time, allowing for flexibility in each) would grow the pool of possible options when adopting a waveless system, and offer insights into optimality considerations.

- The conclusions reached in this paper are based off of previously published literature and a case study of a single implementation of a waveless picking system. It would be of value to generalize the results on system control and optimal choices of management levers by doing a study involving a variety of case studies from various waveless picking systems.

6.5 Epilogue

The Amazon Fulfillment Engine served as a great demonstration of the possibilities which are available to warehousing from adoption of a waveless picking system. The case-study analyzed in this paper also provided significant insight into the gains of migrating a WIP-controlled system over to a hybrid system that can manage system capacity to protect against chute overflow, but that allows additional control over cycle time and thus customer experience.

Finally, after analyzing system performance and response to shipment limiter and demand window adjustments, short-term and long term solutions were put in place to achieve the desired throughput gains. Short-term mechanical adjustment and process improvements revealed the opportunity for a 12.6% increase in immediate throughput capacity, while simultaneously shifting the bottleneck off of operators and system inefficiencies to the sorter itself. For a long-term solution, software teams have begun a study of an optimal way to migrate system management over to the recommended cycle-time controlled system, and are looking at rolling out potential solutions in various locations. A follow-on Leaders for Global Operations internship was created and is currently in process, to continue the study of the relationship of throughput to picking algorithm in a waveless system. As such, Amazon.com continues to show industry leadership in innovation, and in relentless pursuit of continual improvement for processes and process management.

Chapter 7: Bibliography

Ackerman, K.B. (1990). *Practical Handbook of Warehousing.* Van Nostrand Reinhold, New York, NY.

Amazon.com (2008). Annual report.

Amazon.com (2010). Investor website, <http://phx.corporate-ir.net/phoenix.zhtml?c=97664&p=irol-faq#14296>.

Baker, P. and M. Canessa (2009). Warehouse design: A structured approach. *European Journal of Operational Research 193,* 425-436.

Bozer, Y.A., M.A. Quiroz, and G.P. Sharp (1988). An evaluation of alternative control strategies and design issues for automated order accumulation and sortation systems. *Material Flow 4,* 265-282.

Bradley, P. (2007, September). Smoothing the waves. *DC Velocity.*

Bragg, S.J. (2003). Analysis of sorting techniques in customer fulfillment centers. Master's thesis, Massachusetts Institute of Technology.

Campbellsville, (2009, April). Amazon.com Crisplant fulfillment center. Personal tour.

Choe, K.I. (1990). Aisle-based order pick systems with batching, zoning, and sorting. Ph.D. dissertation, School of Industrial and Systems Engineering, Georgia Institute of Technology.

Choe, K.I., G.P. Sharp and R.F. Serfozo (1992). Aisle-based order pick systems with batching, zoning, and sorting. *Proceedings of the International Material Handling Research Colloquium,* 389-420.

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

Craighead, C.W., J.W. Patterson, and L.D. Fredendall (2001). Protective capacity positioning: Impact on manufacturing cell performance. *European Journal of Operational Research 134,* 425-438.

Friedman, T.L. (2005). *The World is Flat.* Farrar, Straus and Giroux, New York, NY.

Friedman, D. (2006). New warehouse management technologies beat the old ones. *Supply House Times 49(7),* 64.

Gademann, A.J.R.M., J.P. Van Den Berg, and H.H. Van Der Hoff (2001, May). An order batching algorithm for wave picking in a parallel-aisle warehouse. *IIE Transactions 33(5),* 385-399.

Gallien, J. (2009, July). Associate Professor of Operations Management, MIT Sloan School of Management (Personal Communication).

Gallien, J. and T. Weber (2009, August). To wave or not to wave? Order release policies for warehouses with an automated sorter. Accepted for publication in *Manufacturing & Service Operations Management.*

Gallien, J. and T. Weber (2009, September). Online appendix to: To wave or not to wave? Order release policies for warehouses with an automated sorter. Accepted for publication in *Manufacturing & Service Operations Management.*

Gue, Kevin (2009, March). New warehouse aisle designs. *Industrial Engineer: IE 41(3),* 52

Hinojosa, A. (2006). What is waveless processing and how can it optimize my operation? Technical report, Fortna, Inc., Atlanta, GA.

Hopp, W.J. and M.L. Spearman (1996). *Factory Physics.* Irwin, Chicago, IL.

Hurley, S.F. and S. Kadipasaoglu (1998). Wandering bottlenecks: Speculating on the true causes. *Production and Inventory Management Journal 39(4),* 1-4

Jackson, D. (2005). Managing and scheduling inbound material receiving at a distribution center. Master's thesis, Massachusetts Institute of Technology.

Johnson, E. (1998). The impact of sorting strategies on automated sortation system performance. *IIE Transactions 30(1),* 67—77.

Johnson, E. and T. Lofgren (1994). Model decomposition speeds distribution center design. *Interfaces 24(5),* 95—106.

Johnson, E. and R.D. Meller (2002). Performance analysis of split-case sorting systems. *Manufacturing Service Operations Management 4(4),* 258-274.

Jutras, C. (2009). Verifying the value of lean six sigma programs. *Manufacturing Business Technology 27(6),* 18.

Kator, C. (2007). Hitting the Right Note. *Modern Materials Handling 62(8),* 32-36.

Kevan, T. (2004). Improving Warehouse Picking Operations. *Frontline Solutions 5(5),* 30-35.

Little, J.D.C. (1961). A Proof for the Queuing Formula: $L = \lambda W$. *Operations Research 9(3),* 383-387.

Lodree, E.J., C.D. Geiger, and X. Jiang (2009). Taxonomy for integrating scheduling theory and human factors: review and research opportunities. *International Journal of Industrial Ergonomics 39(1),* 39-51.

Maloney, D. (2004). Double-duty sorting at Land's End. *Modern Materials Handling 59(3),* 55.

Owyong, M. and Y. Yih (2006). Picklist generation algorithm with order-consolidation consideration for split-case module-based fulfillment centers. *International Journal of Production Research 44,* 4529—4550.

Parks, L. (2008). Advantages of a unique sort. *Stores 90(3),* 78.

Patterson, J.W., L.D. Fredendall, and C.W. Craighead (2002). The impact of non-bottleneck variation in a manufacturing cell. *Production Planning and Control 13(1),* 76-85.

Perry, D. (2007). Continuous processing using a sorter. Technical report, Vargo Adaptive Software, LLC., Austin, TX.

Peterson, C.G. (2000). An evaluation of order picking policies for mail order companies. *Production and Operations Management 9(4),* 319-335.

Pohl, L.M., R.D. Meller, and K.R. Gue (2009). An analysis of dual-command operations in common warehouse designs. *Transportation Research Part E: Logistics and Transportation Review 45(3),* 367-379.

Russell, M.L. and R.D. Meller (2003). Cost and throughput modeling of manual and automated order fulfillment systems. *IIE Transactions 35,* 589—603.

Simchi-Levi, David, Philip Kaminsky and Edith Simchi-Levi (2000). *Designing and Managing the Supply Chain.* Irwin McGraw-Hill.

Szkutak, T. (2009, March). Amazon.com investor presentation, Morgan Stanley Technology Conference.

Appendix: Glossary of Terms

AFE – Amazon Fulfillment Engine, a sortation system with automated pre-sort and manuals sort, installed in 2007 in the Lexington, KY fulfillment center for Amazon.com. Designed to handle a wider variety of size, shape, and weight than many other automated sortation systems.

BMVD – Books, Music, Video, and DVD. Product that is typically classified and processed together due to similarities in shape, size, and weight.

Chasing – The productivity-reducing process whereby items are re-sequenced in the scheduler and prioritized to be picked in order to meet a scheduled ship date rather than as a result of some type of underlying optimization algorithm.

Chute – A physical location where items within a single order accumulate after sortation, in preparation for packing and shipping. At any instant in time, a chute can be occupied by one and only one customer order, and an entire order belongs to one and only one chute.

Chute-dwell time – Defined as the time interval between when the first item in a customer order is assigned to and arrives at its specified chute, until the time that the order finished accumulating in the chute, and the chute is emptied. Chute dwell time is therefore the order accumulation time + waiting time for the completed order until the chute is emptied.

Chute overflow – The phenomenon where there are not any empty, available chutes to accept new openers that flow down from the sorter. The items which cannot be accepted into new chutes must then either recirculate and add congestion to the sorter, or be kicked out of the system for manual processing. Without further, immediate action in a waveless picking system, chute overflow can easily lead to gridlock.

Closer – The final item which is picked for a given customer order of two or more items, completing the picking process for that order.

Critical pull – The time at which all items for all orders due on the next outbound truck need to be picked in order to make it to the shipping dock in time for customer experience.

Customer experience – Literally, the experience a customer has with the fulfillment process. Usually used synonymously with on-time shipping of a customer order to ensure on-time arrival of the shipment to the customer. Can also include quality concerns such as damages, missing items from a shipped order, wrong items shipped, etc.

Demand window – A method for managing and enforcing limits on pick cycle time. The demand window could possibly be implemented as the maximum allowable pick cycle time, as a target average pick cycle time across all shipments, or as some kind of tiered prioritization scheme to achieve a specified variability in pick cycle time around the mean.

FIFO – First-In-First-Out. A type of queuing strategy where items are released from the queue strictly in the order of arrival.

Gridlock – The phenomenon where, due to congestion on an automated sorter, chutes are completely filled and the very items (closers) which could be used to clear occupied chutes cannot reach those same chutes because of the over-congested system. Gridlock is almost always a process-halting problem, and in some systems can take significant time and effort to get the sorter flowing smoothly again.

Inbound – The receiving and stowing process in a fulfillment center whereby product is brought in to the warehouse and placed in inventory for future processing in outbound operations.

Incomplete chute – A chute which has been assigned to a customer order and has only accumulated some but not all of the items necessary to complete the order.

Induct – The process of taking totes filled with many items for many different customer orders, and separating them into distinct, individual trays, for introduction onto the automatic sorting system. Induct is usually a manually process preceding an automated process. Induct can refer to the process itself or to the workstation where induction occurs.

Lateness window – A method for managing and enforcing pick times to meet customer experience. The lateness window setting can be thought of as the minimum processing time needed between the closer pick time of an order and ship time, in order to meet customer experience. Enforcing the lateness window has no real effect on pick cycle time, as it only limits when the last item in a customer order must be picked, regardless of when the opener was picked.

Lean – A process improvement methodology, attributed to Taiichi Ohno and the Toyota Production System, where the focus is on removing "waste" (waste of overproduction, transportation, inventory, motion, defects, over-processing, and waiting) from the system in order to achieve greater throughput, decreased cycle times, and increased productivity.

LEX1 – The call sign for the Lexington, KY fulfillment center for Amazon.com. LEX1 does fulfillment on sortable BMVD and TEKHO, and is one of the largest (in throughput) fulfillment centers in Amazon.com.

Middler – Any item within a customer order that is not an opener or closer.

Off-Peak – The lower volume throughput months of the year. For LEX1, off-peak is the non-Christmas months (Jan-Aug or thereabouts for outbound operations).

Opener – The first item picked in a customer order that consists of two or more items.

Order accumulation time – The time difference between the arrival time of the opener and the closer for a customer order. *See chute-dwell time.*

Outbound – The process whereby items are picked from inventory, sorted to customer orders, and shipped to the customer.

Packable chute – A chute that has completely accumulated all items in a customer order. In the Amazon Fulfillment Engine, a chute is not packable until both all items have been accumulated, and the order has dropped from the incomplete chute into the packable chute.

Packer – Labor associate who removes shipment contents from completed chutes, and packs them into a single box for downstream processing and shipping.

Peak – The high-demand months of the year, typical in retail. Common peak times are the summer months (June, July, and August), and/or the Christmas season (somewhere within the Aug-Dec timeframe).

Pick – Can be used to refer to both the action of picking an item (making a pick), as well as the item scheduled to be picked itself (the opener is the first *pick* in an order)

Pick cycle time – The time difference between the pick time of the opener and the pick time of the closer for a given order.

Picker – Labor associate who picks items from inventory and sends them on to downstream outbound processing.

Process Path – Any one of various possible processing destinations for a customer order. Separate process paths may exist for certain types of items (DVDs versus TEHKO), or even for different sizes of orders.

Rebin – The process and workstation where final sorting occurs. Often used in reference to manual final sort only.

Scheduler – The software algorithm that organizes picks, and prioritizes them based on some underlying model to optimize based on productivity, cycle time, safety, and/or other factors.

SCV – The squared coefficient of variation, a measure for how "in control" a process is. A rule of thumb for using the SCV of process outputs is <0.667 is in control, between 0.667 and 1.33 is moderately controlled, and > 1.33 is an out of control process (Hopp 1996).

Shipment cycle time – The time between the pick time of the opener and the time the shipment is packed out of a chute.

Six-sigma – Process management strategy developed by Motorola in 1981 which focuses on reducing process variability and errors to within prescribed limits.

Sortation – The outbound warehousing process whereby individual items are inducted onto a system, and sorted into holding cells containing individual orders. Can also refer to the system that is performing the sorting process.

TEHKO – Toys, Electronics, House, Kitchen, and Office. Basically a reference to all sortable items (items of small enough size, shape, and weight to fit onto the sortation system) that are not BMVD.

Tote – The container used by pickers to accumulate a number of consecutive picks (usually from 1-20, but literally dependent upon how many of the scheduled picks fit in a tote). Totes are the container used to carry items from inventory to the sortation area, where items are removed from totes in the induct process.

Wave picking – The pick process where a limited number of customer orders are grouped together, then all items within that group are picked and completely contained within a finite number of totes. The term "wave" picking comes from the so-called waves of customer orders.

Waveless picking – The pick process where all orders received by a warehouse are simultaneously eligible for picking, and there is no prescribed order for completing a group of orders before beginning to pick other orders.

WIP – Work in process. Used within this paper to refer to either 1) WIP at pick, all the items which have not yet been picked for which the openers to the corresponding orders have already been picked, or 2) WIP at sort, items in chutes which are awaiting arrival of closers