Throughput Optimization of Multi-Agent Robotic Automated Warehouses

by

James DeWitt Stowe

B.S. Biochemistry, Louisiana State University, 2010
B.A. International Studies, Louisiana State University, 2011

Submitted to the MIT Sloan School of Management and the Engineering Systems Division in partial fulfillment of the requirements for the degrees of

Master of Business Administration
and
Master of Science in Engineering Systems

In conjunction with the Leaders for Global Operations Program at the Massachusetts Institute of Technology

June 2016

©2016 James D Stowe. All rights reserved.

The author hereby grants MIT permission to reproduce and to distribute publicly copies of this thesis document in whole or in part in any medium now known or hereafter created.

Signature Redacted

Signature of Author

Signature Redacted

MIT Sloan School of Management Engineering Systems Division
May 6, 2016

Certified by ________________________________

Dr. Bruce G. Cameron
Thesis Supervisor,
Director System Architecture Lab

Certified by ________________________________

Dr. Roy Welsch
Thesis Supervisor,
Eastman Kodak LGO Professor of Management

Accepted by ________________________________

Dr. John N. Tsitsiklis
Professor of Electrical Engineering
IDSS Graduate Officer

Accepted by ________________________________

Maura Herson,
Director of MBA Program, MIT Sloan School of Management
This page has been intentionally left blank
Throughput Optimization of Multi-Agent Robotic Automated Warehouses

by

James DeWitt Stowe

Submitted to the MIT Sloan School of Management and the Engineering Systems Division on May 6, 2016 in partial Fulfillment of the Requirements for the Degrees of Master of Business Administration and Master of Science in Systems Engineering
Abstract

In 2003 Kiva Systems (now Amazon Robotics) introduced a new type material handling automation to the world. The system is based on the principle that the physical infrastructure that contains inventory should be mobile. Kiva achieved this remarkable advancement by employing a fleet of robots to move shelving to human operators. Broadly, these types of systems are defined in the literature as multi-agent robotic systems. Amazon acquired Kiva Systems in 2012 to incorporate the technology into their operations.

The goal of this thesis is to optimize the throughput of warehouses employing multi-agent robotic automation. It is assumed that extracting inventory from the automated system is the limiting factor in maximizing throughput (i.e. downstream process are unconstrained). Two strategies are advocated: 1) performing velocity segregation of inventory within the automation via a bifurcation between fast selling and slow selling inventory, 2) maximizing pick rates through policies that increase worker retention. It will be shown that velocity segregation increases machine efficiency by increasing the efficiency of delivering inventory to human operators. This assertion will be investigated by developing a theoretical understanding of how inventory velocity impacts machine efficiency and simulating different types of stow strategies impact on system efficiency. It is estimated that some stow strategies can increase machine efficiency by as much as 30%. It will also be shown that the number of man-hours worked by inexperienced pickers explains practically all of the variability of aggregate pick cycle times and hence pick rates, which motivates the argument for worker retention. Together, these two modifications are estimated to increase throughput by 10% over current baseline.
Dedication

For my classmates, the Awesome ‘16s. I’m in. Always.

Acknowledgements

This thesis is the product of a lot of grasping and fumbling with helpful people bemusedly pointing me in the right direction. I am deeply indebted to Joanna Hicks, my supervisor at Amazon, who always took time out of her extremely busy schedule to answer my stupid questions and teach me how warehouses actually work. I must also thank my thesis sponsor at Amazon, Brian Donato, whose incisive comments not only gave me direction, but also pushed me to reach big. Daniel Kelly, a mentor at Amazon, provided me with a key insight at a crucial moment and was always generous with his time and humor.

I would be nowhere without my advisors at MIT: Dr. Roy Welsch and Dr. Bruce Cameron. Roy few out to see me and always answered my calls, even outside of business hours. That meant a lot to me. His knowledge was indispensable as I grappled with the data. Bruce’s guidance made this thesis at least twice as good as it would have been. He consistently astounds me with his ability to see deeply into problems, especially when I have explained them poorly.

I must also thank the LGO program for providing me with an exceptional education. It’s a testament to LGO, MIT and my supervisors that I started LGO with none of the mathematical background or manufacturing context I needed to write this thesis.

Of course, there are the intangibles as well. I love my parents for (among other things) teaching me the value of hard work and perseverance, two qualities that I certainly needed to bring this document to completion. Finally, I have to acknowledge my best friends: Sam, Jonathan, and Brian; who remind me of who I am and who I can become.

It should go without saying, but any errors in this document are mine and mine alone.
# Table of Contents

Abstract ....................................................................................................................... 4

Dedication .................................................................................................................... 5

Acknowledgements ...................................................................................................... 5

Table of Figures ......................................................................................................... 9

1. Introduction ............................................................................................................ 13
   1.1 Fundamentals of Warehousing Operations ......................................................... 13
   1.2 Amazon ............................................................................................................. 14
   1.3 Amazon Robotics .............................................................................................. 15
   1.4 The Transition from Labor to Capital in the Warehousing Industry .................. 16

2. Warehousing operations and MAR Operations ..................................................... 17
   2.1 Inventory Definitions ......................................................................................... 17
   2.2 Hierarchical Aggregation .................................................................................. 18
   2.3 Inventory Flow through an Amazon Fulfillment Center ...................................... 19
   2.4 Manual Operations ........................................................................................... 21
   2.5 MAR Operations ............................................................................................... 22
   2.6 Pick Rate ........................................................................................................... 23
   2.7 The relationship between cycle time and operator utilization ......................... 24
   2.8 Results .............................................................................................................. 27

3. Theory ..................................................................................................................... 29
   3.1 Pile-on .............................................................................................................. 29
   3.2 Quantity Pile-on vs Unique SKU pile-on .......................................................... 30
   3.3 Pile-on, Operator Utilization and Bot Utilization .............................................. 32
   3.4 SKU Velocity, Rack Velocity and Pile-on – A Binomial Model ............................. 36
      3.4.1 Intuition ...................................................................................................... 36
      3.4.2 Useful Properties of Binomial Processes ..................................................... 37
      3.4.3 Model ........................................................................................................ 38

4. Simulation ............................................................................................................... 43
   4.1 Choice of Methods and Literature ................................................................... 43
      4.1.1 Choice of Methods .................................................................................... 43
      4.1.2 Literature ................................................................................................. 44
   4.2 Inputs ................................................................................................................. 45
      4.2.1 Binomial Velocities ................................................................................. 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.2 SKU Spread</td>
<td>48</td>
</tr>
<tr>
<td>4.2.3 Rack Diversity</td>
<td>51</td>
</tr>
<tr>
<td>4.2.4 Mission Time</td>
<td>52</td>
</tr>
<tr>
<td>4.3 Data Structures and Algorithm</td>
<td>54</td>
</tr>
<tr>
<td>4.3.1 Data Structures</td>
<td>56</td>
</tr>
<tr>
<td>4.3.2 Simulation</td>
<td>57</td>
</tr>
<tr>
<td>4.4 Simulation Results</td>
<td>58</td>
</tr>
<tr>
<td>5. Optimization</td>
<td>61</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>61</td>
</tr>
<tr>
<td>5.1.1 The Strategies</td>
<td>61</td>
</tr>
<tr>
<td>5.1.2 Baseline</td>
<td>62</td>
</tr>
<tr>
<td>5.2 Diversity</td>
<td>64</td>
</tr>
<tr>
<td>5.2.1 Mechanism</td>
<td>64</td>
</tr>
<tr>
<td>5.2.2 Predictions</td>
<td>66</td>
</tr>
<tr>
<td>5.3 Segregation</td>
<td>68</td>
</tr>
<tr>
<td>5.3.1 Mechanism</td>
<td>68</td>
</tr>
<tr>
<td>5.3.2 Predictions</td>
<td>69</td>
</tr>
<tr>
<td>5.4 Scatter</td>
<td>72</td>
</tr>
<tr>
<td>5.4.1 Mechanism</td>
<td>72</td>
</tr>
<tr>
<td>5.4.2 Predictions</td>
<td>72</td>
</tr>
<tr>
<td>5.5 Conclusions</td>
<td>74</td>
</tr>
<tr>
<td>6. Cycle Time</td>
<td>75</td>
</tr>
<tr>
<td>6.1 Introduction</td>
<td>75</td>
</tr>
<tr>
<td>6.2 Structure</td>
<td>77</td>
</tr>
<tr>
<td>6.3 The Mechanistic Model</td>
<td>77</td>
</tr>
<tr>
<td>6.3.1 Approach</td>
<td>77</td>
</tr>
<tr>
<td>6.3.2 Data</td>
<td>78</td>
</tr>
<tr>
<td>6.3.3 Results</td>
<td>80</td>
</tr>
<tr>
<td>6.4 Learning Curves</td>
<td>81</td>
</tr>
<tr>
<td>6.4.1 Employee Experience</td>
<td>81</td>
</tr>
<tr>
<td>6.4.2 Model</td>
<td>85</td>
</tr>
<tr>
<td>6.4.3 Diagnostics</td>
<td>85</td>
</tr>
<tr>
<td>6.4.4 Anecdotal Evidence</td>
<td>86</td>
</tr>
</tbody>
</table>
Table of Figures

Figures

Figure 1.1: Basic operations of an Amazon warehouse ................................. 14
Figure 2.1: Hierarchical aggregation illustrated ........................................... 18
Figure 2.2: Value stream map of Amazon warehouse ..................................... 19
Figure 2.3: Basic nomenclature of MAR operations ..................................... 22
Figure 2.4: Activities on a bot mission .......................................................... 25
Figure 2.5: Illustration of the pick and gap cycle. Note that gaps are a regular part of operation as the transition between two racks introduces a small gap. Not to scale. ......................... 26
Figure 3.1: Plan view of a station with racks queued for the picker. Numbers on racks indicates the number of unique SKUs to be extracted from the rack. The grey arrow indicates the flow of racks past the station......................................................... 29
Figure 3.2: Markov chain model of unique SKU picks in front of a picker. λ1 represents the arrival rate of racks with a single unique SKU pick, λ2 represents the arrival rate of racks with two unique SKU picks, and μ is the rate at which the picker performs a pick. In our model λ1+ λ2 = λ , a constant .................. 32
Figure 3.3: Probability of unique SKUs picked per station visit.......................... 33
Figure 3.4: Probability of states given different relationships between λ and μ and allowing λ2 to vary. In the middle plot, the curves for P1 and P0 overlap.................................................. 34
Figure 3.5: OU as a function of unique SKU pile-on. In this figure λ = μ .............. 35
Figure 3.6: Bots utilized as a % of the scenario where λ2 = 0 with λ held constant. In this figure λ = μ .... 35
Figure 3.7: Binomial representation of demand frequency for various types of SKUs. White bars represent no demand arriving for that SKU, colored bars represent demand. ...................... 36
Figure 3.8: Spread distribution of SKUs across racks. .................................. 37
Figure 4.1: Log Log SKU Sales Histogram ....................................................... 46
Figure 4.2: Log Log plot of binomial velocities with the number SKUs that possess that velocity. Data drawn from a 24 hour period of pick data. N ~ 150,000. .................................................. 47
Figure 4.3: Fictitious binomial velocity distribution ....................................... 48
Figure 4.4: Combined SKU spread ................................................................ 49
Figure 4.5: SKU spread based on velocity ....................................................... 50
Figure 4.6: SKU diversity on Racks ................................................................. 51
Figure 4.7: Probability density of SKU diversity on racks. Actual vs fitted distribution. .................................................. 52
Figure 4.8: Mission times compiled over a day from one Amazon FC. N ~ 70,000 .......................................................... 53
Figure 4.9: Mission times for three representative bots. By utilizing mean mission times we may synch the amount of time it takes to perform one mission with our demand simulation intervals ........................................... 54
Figure 4.10: Data Structures ......................................................................... 56
Figure 4.11: Simulation Algorithm ................................................................. 57
Figure 4.12: Actual unique SKU pile-on vs simulated .................................... 58
Figure 4.13: Q-Q Plot of Simulated Vs Actual Pile-on ..................................... 59
Figure 4.14: Actual Unique SKU pile-on versus simulated with different stow strategies .................................................. 60
Figure 5.1: Illustration of random stow strategy ............................................ 62
Figure 5.2: Simulated random stow pile-on .................................................... 63
Figure 5.3: Fictitious Distribution Simulation compared against Empirical Simulated and Actual .................................................. 63
Figure 5.4: Q-Q Plot showing fictitious dist. simulation compared to empirical dist. simulation ........................... 64
Tables

Table 3.1: Example pick list.................................................................................................................... 31
Table 3.2: Pile-on for each rack.............................................................................................................. 31
Table 3.3: Unique SKU pile-on for each rack........................................................................................ 31
Table 4.1: Simulation Inputs .................................................................................................................. 45
Table 4.2: Fictitious binomial velocity distribution. ............................................................................ 48
Table 5.1: Summary Statistics of baseline .......................................................................................... 63
Table 5.2: Mean pile-on for different bifurcation thresholds.............................................................. 71
Table 5.3: Scatter simulations mean pile-on ........................................................................................ 74
Table 6.1: Regressors and response descriptions for cycle time modeling........................................ 79
Table 8.1: Baseline model, one warehouse. The two cases indicate a 10% increase in throughput and a
15% increase in throughput respectively, ............................................................................................ 101
Table 8.2: Financial model, two warehouses; one robotic, one manual.............................................. 102
This page has been intentionally left blank
1. Introduction

This thesis attempts to answer the question: how does one optimize throughput of an automated warehouse under the assumption that pick is the bottleneck on outbound volume? We discuss a very specific type of automation: that of multi-agent robotic systems (or MARs). The most highly developed and commercially successful warehouse automation based on MARs was created by Kiva Systems in 2003. Kiva was subsequently purchased by Amazon and rebranded as Amazon Robotics. The author had the good fortune to work for six months in an Amazon fulfillment center that employed MAR technology.

The approach advocated for in this thesis boils down to two recommendations. First, by organizing inventory based on velocity (sales/time) within the automated storage system, the system becomes more efficient at delivering inventory to human operators for extraction. Second, this thesis will show that aggregate human pick rates varies inversely with the average number of hours worked by inexperienced workers. We will argue that both of these solutions must be employed in tandem to reach a sustainable increase in pick rates and, hence, throughput.

The purpose of this chapter is to motivate the discussion, introduce the warehousing industry, place Amazon in the context of that industry, and discuss the shift of warehousing operations from labor to capital intensive processes.

1.1 Fundamentals of Warehousing Operations.

In common language, a warehouse is a space where inventory is stored. From the perspective of supply chain design, a warehouse is a buffer: a place where work in progress (WIP) inventory or finished goods may be stored for an indefinite period of time. Inventory storage optimizes supply chain operation by hedging against uncertainty in the market and decreases logistics costs by allowing bulk transportation of inventory to specific geographic regions.

By our definition, warehousing operations do not include any transformative steps (i.e. inventory passes through processes without changing form). Hence internal operations
consist of two broad functions: inbound and outbound (see Figure 1.1). Inbound processes unload inventory from transportation equipment, prepare it for storage, and store it into a storage system. (We define a storage system as both physical infrastructure and the record keeping mechanism that tracks the location of inventory units). Outbound processes extract units of inventory from the storage system, prepare them for shipment, and load them onto transportation equipment.

In a market where the inventory in storage is in high demand (hence requiring high throughput), two aspects of these process paths are central to well-functioning warehousing operations: 1) High fidelity of the records used to track inventory location and 2) the speed at which units can be extracted from the storage system, packed, and shipped.

1.2 Amazon

Amazon is a technology company founded in 1995 by Jeff Bezos. It has businesses that span retailing, consumer electronics, entertainment, and cloud computing [1]. Revenues in fiscal 2014 amounted to approximately $90 Billion with a gross profit of $26 Billion. For the past three years Amazon's net income has fluctuated between a gain and loss of less than a percentage point of total revenue [2]. YOY revenue growth has ranged between
20% and 41% over the past 8 years [3]. In 2015, Amazon became the most valuable retailer in the United States as measured by market capitalization. This paper focuses exclusively on the retail arm of Amazon’s business and the remainder of this section concerns the same.

Amazon was founded as an online bookstore and has since expanded into a diverse and growing mix of product lines. The basic value chain operates in the following way: A customer selects products they desire on their website along with desired shipping speed and payment options. Amazon’s backend IT infrastructure allocates the orders to fulfillment centers (warehouses or FCs) in their network. Electronic instructions are relayed to the FC’s operations team which finds the units of inventory to be shipped, packages them, and places them into the logistics network for delivery to the customer.

The value proposition Amazon offers customers is multifaceted: an open forum to gather information on products, an extremely wide selection, low prices, high probability of on-time delivery, and accommodating customer service. Much of the value proposition depends on a warehousing operation that can stock, allocate, track, store, and move a large number of SKUs with a high degree of reliability at low cost.

In 2012 Amazon acquired Kiva Systems for $775 million [4]. Kirshner reported in 2012 that the acquisition of Kiva by Amazon served two strategic objectives: To gain access to the proprietary software that operates their automation and to have a controlling ownership in Kiva before integrating the technology into their operations.

Kiva Systems was founded in 2003 by Mick Mountz, Peter Wurman and Raffaello D’Andrea [5]. The founders built the company on the premise that the traditional model of warehousing operations – people searching through storage to find and move inventory – was obsolete. Instead, they envisioned an automated system where people remained stationary and inventory moved to them. In theory, this innovation would decrease the number of non-value added operations labor performed in the warehouse (i.e. transiting from shelf to shelf) and hence increase the amount of value added operations labor could perform (picking and packing). As labor hours account for a large portion of the total cost
of warehousing operations, the founders assumed that such a technology would be attractive to high throughput distributors.

The implementation that Kiva developed relies on a fleet of robots that carry mobile shelving to stations where human workers load or extract inventory from the system. Though the author will attempt to describe the workings of the system in Chapter 2, video provides an excellent introduction. The author recommends the video referenced here\(^1\).

The software that orchestrates order allocation and robot movement represents the fundamental achievement of Kiva technology [5]. This software relies on a programming paradigm called multi-agent programmatic systems [6]. Since Kiva's inception several competitors based on similar models have arrived [7] [8].

In 2015 Kiva Systems was rebranded as Amazon Robotics [9].

1.4 The Transition from Labor to Capital in the Warehousing Industry

Two fundamental questions lie at the heart of this thesis. First, how can companies utilize inventory sortation to optimize warehousing operations given the new constraints automation imposes? Second, how does automation change the nature of work for both line workers and managers – and what does this imply about how they should be managed?

Fundamental to these questions is the author's belief that we are in a period where foundational assumptions about operations are changing. It is taken as an axiom among operations professionals and academics that labor processes are more flexible than automated processes. The advent of Kiva, and the design paradigms it embodies, challenge this principle. In physical space, MARs are modular, versatile, and extensible: the constraints on their function dominated by software rather than physical layout and dynamics. This approach lends itself to rapid adaptation in physical space. If borne out, these observations could have wide ranging implications for the way businesses and society functions.

\(^1\) https://www.youtube.com/watch?v=lWsMdN7HMuA
2. Warehousing operations and MAR Operations

2.1 Inventory Definitions

For the purposes of the rest of the paper the following inventory terms are defined abstractly and by way of example.

1) Unit – A single item of inventory. This is the most granular sub-classification of inventory. Depends on how the product is marketed to customers. The number of units a retailer has on hand determines the number of customers they can service with available inventory. The word “each” is equivalent to unit.

2) SKU – Stock keeping unit. More informally – refers to a sub-classification of inventory in which units are interchangeable with each other for the purposes of sales.

Note: Expiration dates cause us some difficulties with this definition because, in reality, two lots of the same SKU with different expiration dates are not technically equivalent in terms of sales. But, we note this difficulty and ignore it. It does not bear strongly on the discussion that follows.

For example imagine we intend to sell cans of soda at a sports game, say Coke and Sprite. One six pack of Coke represents a single SKU because all of the cans contained in the package are “equivalent” from the customers perspective. The customer doesn’t care which can of Coke they get, they only care that they get a can of Coke. The cans of Coke are units of inventory. They are essentially indivisible.

Extending the example a bit further allows us to see how the definition of unit depends on marketing. If instead one wanted to market six packs of Coke, rather than cans, then a unit is now a six pack. Hence the definition depends directly on what’s on offer to the customer.

One further term that will be of use later on:

3) Diversity – The number of SKUs on offer to customers. Retailers with larger diversity relative to their competitors have a greater number of SKUs to offer customers.
2.2 Hierarchical Aggregation

In supply chain operations, the concept of hierarchical aggregation describes a way that units are organized in physical space and databases. In space, units may be contained in a case or container. Returning to our soft drink example where we are marketing cans: The cans that contain the Coke are primary packaging. They contain a single unit of inventory. The plastic that binds cans together in a six pack contains six units. The pallets that carry the six packs during transport contains some number of six packs and so on. Hierarchical aggregation may be visualized as a tree structure with units as the leaves and each parent node representing a container that holds all of its children (see Figure 2.1). In the jargon, a child node is said to be "bound" to its parent. Again, returning to the soft drink example, each can in the six pack is bound to the six pack itself, which is a new level of aggregation. Similarly, the six packs are bound to the pallet on which they were transported.

![Figure 2.1: Hierarchical aggregation illustrated.](image)

The data structure that tracks the location of units within Amazon inventory mirrors the physical tree structure of containers that hold units. Much of the backend infrastructure that powers retail operations are devoted to tracking transactions as units transition through different container types. A unit that enters the Amazon system remains bound to a container throughout its lifecycle. For example, a unit usually enters the Amazon
network bound to a case. That unit will then be removed from its case (unbound) and bound to a bin. When the unit is picked it will be unbound from the bin and bound to a tote. Then it will be transported to packing operations where it will be unbound from the tote and bound to a box. In physical space, these transactions consist of two operations: movement of the physical item from one container to another and scanning barcodes of the unit and the two containers to update the tracking data structure.

Hence there are two trees that ideally mirror each other. The physical tree of containers that Amazon uses to organize and transport its inventory, and the data tree that allows the IT infrastructure to pinpoint which container holds which units. In reality, the two trees are never in perfect alignment and resources are devoted to maintaining the accuracy of the data tree.

Taking a highly abstract view, one may understand much of Amazon operations through the lens of container transactions. Pushing the concept further, one may understand process improvement at Amazon as making container transactions as cost efficient as possible in physical space.

2.3 Inventory Flow through an Amazon Fulfillment Center

In this section we examine the steps a unit of inventory transverses through an Amazon warehouse (Figure 2.2).

![Value stream map of Amazon warehouse.](image)

*Figure 2.2: Value stream map of Amazon warehouse.*
1) Receiving

Inventory enters the FC by way of loading docks where pallets or other large containers are unloaded from tractor trailers. The pallets are scanned at the loading dock to unbind the pallets from the tractor trailer and bind them to the FC. The pallets are disaggregated into cases or totes. Totes are standardized, general purpose plastic containers Amazon uses for unit organization and transport.

If mixed pallets arrive (pallets holding multiple SKUs), the pallets are disaggregated before the cases are bound to the FC. Ideally, units are bound to the FC at the highest level of aggregation possible. If a case contains one SKU, that case is scanned and all of the units contained in the case are bound to the FC (License plate receive). If a case contains multiple SKUs, it must be disaggregated and the units scanned individually.

2) Decant

Often cases or totes move directly from receive to the stow process, but decanting is a possible intermediate step. Decanting is the process of unbinding received units from the cases or totes in which they arrived and binding them to a large cart container. Moving the units to the cart may speed the stow process.

3) Stow

The stow process unbinds received units from their containers and binds them to bins. Bins are the primary storage infrastructure Amazon employs to hold inventory.

4) Pick

The pick process selectively unbinds units from bins based on customer demand and aggregates the units into orders. For example, a customer might order three different SKUs in different quantities that are distributed randomly throughout the storage bins. An ideal pick process output should be a single container that holds the correct SKUs in their respective quantities.
5) **Pack**

The pack process consolidates the customer order into as few shipping containers as possible and finalizes the preparation of the containers for shipment. In a typical Amazon fulfillment center the shipping containers are cardboard boxes and the pack process consists of placing the units into the boxes, adding filler to prevent the units from moving during transportation and sealing the boxes with tape.

6) **Ship**

The ship process involves loading the boxes bound with completed orders into tractor trailers in preparation for shipping to tertiary facilities for distribution to the postal network.

Now that we have some basic understanding of the process in the abstract we turn to a discussion of how these steps are performed in two different models: manual and MAR operations. As of now, manual and MAR operations differ in stow and pick; hence we will focus primarily on these two steps.

2.4 Manual Operations

Manual storage space consists of a set of long linear racks affixed to the ground. Each shelf on each rack is divided into bins of various sizes. Each bin has a unique barcode identifier.

Stowing into this system consists of a worker wheeling a cart loaded with inventory into the field, searching for open space, finding open space and transferring inventory from the cart to the shelving. The inventory management system registers inventory as for sale when it is placed and bound to a bin on the racks.

When an item of inventory is bound to an order and marked to be extracted from the racks, the inventory item, along with its location, is placed in a queue for one of the pickers. When the picker is assigned to pick a unit from the queue, they locate the unit in the racks, unbind it from the bin (hence marking it as picked) and bind it to a container that will be used to transport it downstream.
To summarize and simplify the above discussion, storing and extracting inventory from this system is completely dependent on people. Hence, outbound volume (assuming processes downstream are unconstrained) depends completely on headcount. We can think about throughput with the following equation.

\[ Volume = \text{Head Count} \times \text{Length of Work Week} \times \text{Rate} \]

Hence management can use headcount as a lever to move volume. There are secondary effects that decrease the marginal productivity as workers start to flood the aisles of the storage space (congestion), but these effects can be effectively ignored or mitigated.

2.5 MAR Operations

A MAR storage space consists of many modular racks that are square in the length and width dimensions. These racks are not affixed the ground and are meant to be moved from place to place within the bounds of the inventory field. (Nomenclature is explained in Figure 2.3)

![Figure 2.3: Basic nomenclature of MAR operations.](image)

The field is surrounded by stations where workers interact with the inventory. To stow items in the field, inventory is brought to stow stations and robots (or bots) bring racks to the stations based on how much space they have available and the types of bins that they
have (there are specialty bins for certain types of inventory, like soft lines). Ideally, the correct type of racks are brought to stations without intervention from the operators. The MAR recognizes what types of SKUs are to be stowed at each station and assigns bots to bring the correct racks.

Similarly, when a unit is to be extracted from storage, the system selects which rack to bring to station and assigns a bot to move it. At the pick station a computer indicates to the operator which unit to pick from a specific bin. In the ideal case, the operator locates the unit, scans it, and the system recognizes that the unit has been removed from storage.

Stations are fixed in quantity, hence the maximum volume that can be extracted from the inventory field can be described by the following equation.

\[ \text{Volume} = \text{Hours in a week} \times \text{Station Utilization} \times \text{Number of Stations} \times \text{Pick Rate} \]

Where station utilization is expressed as a percentage and rate is expressed in units/time. Station utilization describes the number of hours each station is being used as a fraction of twenty four hours. Stations have a minimum required amount of idle time for shift changeovers and maintenance, therefore this number can never approximate 100%. The number of stations available can be considered fixed because substantial capital investment in bots is necessary to build a new station. Hence, the only way managers can substantially impact maximum throughput is thorough changes in pick rate.

2.6 Pick Rate

Pick rate, the number of units extracted from the robotic inventory field at a station per unit of time, can be described by the following equation:

\[ \text{Pick Rate} = \frac{3600 \times \text{Operator Utilization}}{\text{Cycle Time}} \]

This expression shows the fundamental dependence of pick rate on the operators (cycle time) and the machines (operator utilization or OU). Cycle time is how quickly an operator can perform a pick and has units of seconds. OU is the percentage of time that a rack is presented to an operator. In this formulation pick rate is expressed in units/hour. These units are arbitrary, but convenient for the purposes of management. For ease of discussion,
the rest of this paper will use the units presented here and no effort will be made to make the calculations dimensionless.

The goal of our attack is to maximize pick rate. If we may assume that operator utilization and cycle time are independent, then the problem statement is simple:

*Maximize pick rate by minimizing cycle time and/or maximizing operator utilization.*

Since we are working in an environment with constrained resources, logically we want to maximize the use of those resources. Hence we may reasonably ask which lever, cycle time or operator utilization, has the strongest effect on pick rate. By differentiating pick rate with respect to cycle time and OU, we see that cycle time has an inverse quadratic effect on a change in pick rate while OU is constant. OU also suffers from the disadvantage that a well-designed system will have a value close to the theoretic maximum - 100%. We may conclude that cycle time is the only effective lever to move pick rate and hence throughput.

But our assumption about the independence of OU and cycle time is wrong. We devote the next section to discussing how the two are interrelated.

2.7 The relationship between cycle time and operator utilization

The lower the cycle time of a picker, the more inventory a MAR needs to queue in front of that picker to ensure that they remain busy. This results from fact that pick cycle times are much less than the mean length of a mission. In the simplest case, imagine that every rack brought to a station only contains one SKU that needs to be extracted for an order. The pick rate then correlates exactly with the number of racks that need to be brought to a station. But, cycle time is a random variable with some mean and variation. For the MAR system to maximize the "up time" (or operator utilization) it must maintain a queue in front of the operator to buffer them against a string of very fast picks, which would leave them in a gap situation: unutilized until the MAR brings them more racks.

We may give an intuition about the relationship between cycle time and the number of bots allocated to a station with a discussion of mission time.
Figure 2.3 outlines the basic operations a bot must perform in one complete mission. When a rack is allocated to a station, its assigned bot must fetch the rack from its location, deliver it to the station, queue in front of the station, have its inventory extracted by the operator and then return the rack to a storage location. When the operator is interacting with the rack, the rack and bot are said to be sitting on the action fiducial. This complete cycle is termed a mission and the mean amount of time it takes for all of the bots in the field to perform these operations is the mean mission time. The relationship between the number of bots allocated to a station and picker cycle time may be stated:

\[
\text{No. Bots Allocated to Station} \approx \frac{E[\text{Time a bot stops on the action fiducial}]}{E[\text{mission times}]}
\]

And:

\[
\text{Picker Cycle Time} \approx E[\text{Time a bot stops on the action fiducial}]
\]

We see that for each rack brought to a station, a bot must be allocated. If the pickers work faster (on average) more bots are engaged in total to keep the pickers buffered and busy.

---

2 This terminology arises from the QR codes that the robotic system uses to locate points in the field. Each inventory field is laid out with a precise grid pattern of QR codes that the bots track with a bottom facing camera. The QR codes are termed fiducials, hence the origin of the term action fiducial.

3 The transition times of one bot cycling away from the action fiducial and another replacing it is included in our definition of cycle time. This interval is on the order of 1/10 of a mean pick cycle time.
Hence we run into a hard constraint: the number of bots active in the field. This constraint is observed in practice. In periods of high demand, operator utilization degrades because pickers work at elevated rates, more stations are utilized, and hence the field demands more bots than are available. This fundamentally results from the fact that there are not enough bots in the field to meet all of the demand. Operator utilization degradation is illustrated in Figure 2.5.

---

![Diagram of pick and gap cycle]

**One pick and gap cycle**

Productive time (pick):
- registering what product is to be picked,
- searching for product,
- placing product in tote

Unproductive time (gaps):
- rack transitions
- rack missing from station

**Ideal sequence of picks and gaps**

---

**Pick and Gap sequence where drive demand exceeds supply**

---

*Figure 2.5: Illustration of the pick and gap cycle. Note that gaps are a regular part of operation as the transition between two racks introduces a small gap. Not to scale.*

The solution to this problem seems simple: add more bots. This is non-optimal for two reasons. First, bots are the largest capital expense in building a MAR. Each bot added to the field increases the fixed costs of building the field and hence negatively impacts the NPV of building the facility. Second, it's not an effective solution to the problem. As more bots are added to the field, congestion effects increase mean mission times and hence
retard the benefit. The shape of this function has not been well studied, but it has been observed.

We restate our objective given a constraint and our new understanding that OU and cycle time are linked:

Maximize pick rate by minimizing cycle time and maximizing operator utilization without increasing the number of bots in the field.

Our attack on the problem is two pronged. First, we must alleviate the OU constraint on cycle times (that as cycle times decrease, OU degrades). Second, we must decrease cycle times. The order is important. A decrease in cycle times cannot happen in the absence of changing OU's dependence on cycle times.

The OU problem is fundamentally an issue of machine efficiency, the cycle time problem one of human learning. The results are summarized below, the justification for those results are explained in the chapters that follow.

2.8 Results

On the subject of OU degradation, this thesis will argue that a velocity stow process will increase throughput while holding the number of bots in the field constant when compared to a random stow strategy. Velocity stow is the practice of organizing SKUs based on the rate at which they sell (velocity) and stowing them into segregated racks. Random stow is the practice of stowing inventory into racks without any prior organization. In plain language, velocity stow means only stowing fast selling inventory in a sub-population of racks and only stowing slow selling inventory into a separate, disjoint population of racks. The question of how to define "fast selling" occupies the OU chapters below.

For cycle time, the data indicate that work force experience (WFE) overwhelmingly determines mean cycle times, which suggests maximizing worker retention as a way to minimize cycle time. Technical analysis will help us in understanding this driver of cycle time, but the prescriptions will be of a softer character. Much of this discussion will focus on learning curves and what motivates people to come to work.
The rest of this paper will broadly follow two tracks. The following three chapters (theory, simulation and optimization), will attack the problem of OU and system machine efficiency. The two chapters that follow that (cycle time and turnover) will tackle the issue of cycle time. We will circle back to all issues in the conclusion.

As we move away from the introductory matter of this thesis, it is worthwhile to step back and call attention to a broader theme. The two sub-solutions to the problem under discussion, (throughput maximization of a MAR system) highlight the intimate interaction of man and machine as industry shifts to a higher automation paradigm. MARs are a dramatic leap forward towards a completely automated warehousing facility; indeed, business models reliant on the next generation of automation are already seeking funding [10]. Until that time comes though, operations are still based on the three pillars of people, process and technology. As managers and professionals, we ignore any of these aspects, and their interactions, at our own peril.
3. Theory

3.1 Pile-on

This chapter will explore in depth the concept of pile-on, the relationship between pile-on and operator utilization, and the relationship between SKU velocity and pile-on. While on this line we will introduce major assumptions that will be critical to interpret the results presented in Chapter 4.

![Diagram of a station with racks queued for the picker. Numbers on racks indicate the number of unique SKUs to be extracted from the rack. The grey arrow indicates the flow of racks past the station.]

Figure 3.1: Plan view of a station with racks queued for the picker. Numbers on racks indicates the number of unique SKUs to be extracted from the rack. The grey arrow indicates the flow of racks past the station.

Pile-on is the mean number of units picked out of a rack during a single rack visit. For example, five racks pass through a station, the picker upon each visit picks 1 unit, 1 unit, 1 unit, 2 units, 1 unit (see Figure 3.1); the pile-on for these visits is just the number of total units extracted divided by the number of racks it took to bring those units: \(6/5 = 1.2\). In order to calculate a pile-on we need these two things: number of rack visits in a period of time and the number of units picked during that same period. We can easily see that pile-on is a measure of the efficiency of the system. All else equal\(^4\), an inventory field with

\(^4\) Meaning that the demand for SKUs and the workforce operating the stations are equivalent.
higher pile-on will have required less racks to be brought to station to achieve the same amount of unit extraction. We also can see that the situation with higher pile-on requires fewer bots for two reasons: 1) the system is bringing fewer racks to the stations 2) each rack is staying at the station, on the mean, longer, hence the queues in front of the station can afford to be shorter.

3.2 Quantity Pile-on vs Unique SKU pile-on

When a customer places an order, they do so on two dimensions. They order a certain number of SKUs and a certain quantity of each SKU. This order is then split and distributed to different pickers based on their station's proximity to racks that hold the inventory requested. (Note: Amazon has downstream processes for aggregating orders from different pickers, so there is no need for a picker to pick an entire order for a single customer).

Pile-on also comes in two dimensions: quantity pile-on, where a picker picks the same SKU in quantity, and unique SKU pile-on, where a picker picks multiple SKUs from the same rack. For the purposes of this paper we are exclusively concerned with unique SKU pile-on for the following reasons.

1) Quantity pile-on is less valuable from a cycle time perspective than unique SKU pile-on. Data shows that quantity pile-on has a much lower cycle time per unit than SKU pile on. This seems like it would be a good thing, but part of the benefit of pile-on is that it keeps a rack in front of a station for a longer period of time productively.

2) Concentrating too much quantity of a single SKU on a rack can bottleneck the system for that SKU, because it can only be accessed by one picker at a time. This was observed by engineers at Kiva early on in the development process. If demand for a single SKU is high and it is distributed across multiple racks, the system can parallelize extracting it from the system.

3) It simplifies simulation of the problem in a way that underestimates the benefit of the velocity organization of SKUs. There is a great deal of overlap between fast selling SKUs and quantity picks. In a single day of pick data, ~50% of the SKUs that were picked for quantity, were also picked multiple times. Hence,
optimizing for Unique-SKU pile-on will also pick up a substantial amount of quantity pile-on as well.

4) Simulating quantity picks vastly complicates simulation, because one must implement a system that tracks each individual unit of inventory. Basing the simulation on SKU velocity means we must only track which SKUs are represented on which racks.

To be clear, an example of an order list to be picked is displayed in Table 3.1:

<table>
<thead>
<tr>
<th>Racks in Queue</th>
<th>SKU</th>
<th>QTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACK 1</td>
<td>SKU 1</td>
<td>1</td>
</tr>
<tr>
<td>RACK 2</td>
<td>SKU 2</td>
<td>3</td>
</tr>
<tr>
<td>RACK 3</td>
<td>SKU 3</td>
<td>1</td>
</tr>
<tr>
<td>RACK 3</td>
<td>SKU 4</td>
<td>2</td>
</tr>
</tbody>
</table>

*Table 3.1: Example pick list.

This table indicates that the picker should remove quantity 1 of SKU 1 from rack 1; quantity 3 of SKU 2 from rack 2; and quantity 1 of SKU 3 and quantity 2 of SKU 4 from rack 3. The pile on for each rack would be:

<table>
<thead>
<tr>
<th>Rack</th>
<th>Pile-On</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACK 1</td>
<td>1</td>
</tr>
<tr>
<td>RACK 2</td>
<td>3</td>
</tr>
<tr>
<td>RACK 3</td>
<td>3</td>
</tr>
</tbody>
</table>

*Table 3.2: Pile on for each rack.

The unique SKU pile-on for each rack would be:

<table>
<thead>
<tr>
<th>Rack</th>
<th>Unique SKU Pile-On</th>
</tr>
</thead>
<tbody>
<tr>
<td>RACK 1</td>
<td>1</td>
</tr>
<tr>
<td>RACK 2</td>
<td>1</td>
</tr>
<tr>
<td>RACK 3</td>
<td>2</td>
</tr>
</tbody>
</table>

*Table 3.3: Unique SKU pile-on for each rack.
Hence, quantity picks do not play into our analysis. We are strictly concerned with pile-on resulting from multiple unique SKU picks. From this point forward we use pile-on and unique SKU pile-on interchangeably, except if a distinction is made explicitly.

3.3 Pile-on, Operator Utilization and Bot Utilization

The complete dynamics of how pile-on influences OU are difficult to analyze because much of the sophistication of MARs are their algorithms designed to keep operators busy. Nonetheless, a toy queuing model is sufficient to get an understanding of the gross dynamics of the system. See Figure 3.2.

![Markov chain model of unique SKU picks in front of a picker.](image)

Figure 3.2: Markov chain model of *unique SKU picks* in front of a picker. $\lambda_1$ represents the arrival rate of racks with a single unique SKU pick, $\lambda_2$ represents the arrival rate of racks with two unique SKU picks, and $\mu$ is the rate at which the picker performs a pick. In our model $\lambda_1 + \lambda_2 = \lambda$, a constant.

Shown above is a six state discrete Markov chain representing the number of *unique SKU picks* queued in front of a picker. The selection of six states is arbitrary and stems from the fact that the author was unable to derive a closed form solution for the infinite state case. A better representation of the system is probably an 8 state chain, but for simplicity of graphics the author chose the six state case. It does not matter though, the dynamics for each system are the same regardless of state number.

---

5 In the author's experience, a 3-5 bot queue in front a picker is typical.
In Figure 3.2 there are three variables of interest: \( \lambda_1 \) represents the arrival rate of racks with a single SKU to pick, \( \lambda_2 \) represents the arrival rate of racks with two SKUs to pick and \( \mu \) represents the cycle time of a pick. We note here another simplification of the system presented in Figure 3.2, an accurate representation of the system would include arcs that represented a rack with three unique SKUs to pick, an arc to represent the 4 SKU case and so on. We exclude these cases because each case becomes exponentially less probable according to pick data (Figure 3.3) and it simplifies the mathematics considerably.

![Distribution of SKUs extracted for a station visit](image)

*Figure 3.3: Probability of unique SKUs picked per station visit.*

In our analysis, the failure state is P0. P0 is where there are no picks in front a picker and the picker is unutilized. In fact we can think of OU as

\[
OU = 1 - P0
\]

For convenience and without reference to data, we assume arrivals and cycle times are distributed Poissonically. The analysis then reduces to recursion and algebra. Figure 3.4 and Figure 3.5 were plotted under the assumption that \( \lambda_1 + \lambda_2 = \lambda \) and \( \lambda \) is in various relations to \( \mu \). We see immediately from Figure 3.4 that in all scenarios higher pile-on translates into an increase in the probability that the queue will be in a state that buffers the operator.
Figure 3.4: Probability of states given different relationships between $\lambda$ and $\mu$ and allowing $A_2$ to vary. In the $\lambda=\mu$ plot, the curves for $P_1$ and $P_0$ are coincident.
From Figure 3.5 and Figure 3.6 we see that increasing the ratio $\lambda_2/\lambda_1$ not only increases OU but also decreases the number of drives utilized. (Note, both figures were created under the assumption that $\lambda = \mu$.)

![Operator Utilization (1−P0) as a Function of Pile-On](image1)

*Figure 3.5: OU as a function of unique SKU pile-on. In this figure $\lambda = \mu$.  

![Number of Bots Allocated as a Function of Pile-On](image2)

*Figure 3.6: Bots utilized as a % of the scenario where $\lambda_2 = 0$ with $\lambda$ held constant. In this figure $\lambda = \mu$.

This model is a simplification over the realities of MAR operations, but it allows us to see the basic outlines of an attack on the bot allocation problem. If we increase pile-on, it is
possible that OU will increase (or at least remain steady) and bot utilization will decrease, freeing up more bots to be redeployed.

3.4 SKU Velocity, Rack Velocity and Pile-on – A Binomial Model

3.4.1 Intuition

Before discussing the model in detail, it is worthwhile to outline a basic intuition of how the author chose to parameterize the problem. Our attack consists in simplifying demand for each SKU, which is a continuous stochastic process of undetermined distribution, into a discrete binomial process of one trial - sometimes called a Bernoulli process. In this model, we will ignore quantity as a factor, we are only concerned with if demand arrives for a SKU in any quantity. Hence it is a binary question: did demand arrive in this interval? Yes – 1, No – 0. The proportion of 1s to 0s is determined by a SKUs velocity.

To make this clear, please refer to Figure 3.7.

![Binomial Arrivals of SKUs](image)

*Figure 3.7: Binomial representation of demand frequency for various types of SKUs. White bars represent no demand arriving for that SKU, colored bars represent demand.*

In the figure above, an hour is divided into 60 minute-long intervals. The colored bars indicate if demand has arrived in that interval.

Discrete arrival of demand is the only abstraction necessary to understand the model presented in the next section, but looking slightly ahead to simulation we should discuss rack velocity.
In Figure 3.8 we contend with two further dimensions. First is the variable number of SKUs that each rack carries. In the figure, rack 1 carries 2 SKUs; rack 2, 3 SKUs; and rack 3 only 1 SKU. We must capture this variability because it can range quite dramatically. It’s possible for one rack to be at full cube utilization while only containing one SKU.

The other difficulty is that the number of racks that contain a SKU is dependent on the velocity of that SKU. The high velocity SKU is likely to have more units in the inventory field and hence be spread over more racks.

The issue of pile-on may be parameterized based on these three dimensions: binomial SKU velocity, the SKU capacity of each rack, and the relative representation of that SKU on the population of racks. These parameterizations will be made explicit in Chapter 4. We now turn to understanding how the velocity of SKUs on racks determines pile-on.

3.4.2 Useful Properties of Binomial Processes

For a random variable $X$ distributed according to the binomial distribution with parameters $n$ and $p$ we have:
\[ E[X] = np \]

Now consider two independent random variables binomially distributed: \( X \sim B(n, p) \) and \( Y \sim B(n, q) \). Their sum is, by linearity of the expectation operator:

\[ E[X + Y] = E[X] + E[Y] = np + nq = (p + q)n \]

Similarly, we have for a random function \( X(t) \) distributed according to a binomial process \( \text{BP}(p) \):

\[ E[X(t)] = pt \]

Finally, if we have two independent random functions \( X(t) \sim \text{BP}(p) \) and \( Y(t) \sim \text{BP}(q) \):

\[ E[X(t) + Y(t)] = E[X(t)] + E[Y(t)] = pt + qt = (p + q)t \]

3.4.3 Model

Consider a MAR warehouse in a simplified world in which the following rules apply:

1) Demand for SKUs arrive in arbitrarily sized discrete time intervals – we term these demand intervals.
2) We never get more than one order for the same SKU in one demand interval.
3) We can always define a time interval longer than a demand interval where the probability of demand is steady. We call this the interval of analysis.

The warehouse contains \( N \) SKUs indexed over the integers \( i = 1, 2, \ldots, N \). Similarly, the minutes in the day are indexed \( j = 1, 2, 3, \ldots, 1440 \). Hence a random variable \( X_{i,j} \) describes whether we have demand for SKU \( i \) in minute \( j \) by assuming the value of 1 with probability proportional to the popularity of the SKU, but stable in some interval longer than a minute.

Hence:

\[ X_{i,j} \sim B(1, p_i) \]

Where \( B(1, p_i) \) is the Binomial distribution of one trial with probability of success \( p_i \). This is also called a Bernoulli trial.
SKUs are stowed in $M$ different racks and it would be useful to know at any one time $t$ what the probability is that demand will be generated on a rack for any of its SKUs. If the racks are indexed $k = 1, 2, \ldots, M$ then:

$$B_{k,t} = \sum_{v \in \{i \in \text{rack } k\}} X_{v,t}$$

Where $B_{k,t}$ is a random variable distributed according to a Poisson Binomial distribution that describes how many potential picks are generated on rack $k$ at time $t$. (Demand for an SKU on a rack equates to a potential pick.)

Taking the expectation:

$$E[B_{k,t}] = E\left[ \sum_{v \in \{i \in \text{rack } k\}} X_{v,t} \right]$$

$$E[B_{k,t}] = \sum_{v \in \{i \in \text{rack } k\}} E\left[ X_{v,t} \right]$$

$$E[B_{k,t}] = \sum_{v \in \{i \in \text{rack } k\}} p_i$$

Hence the expected number of potential picks generated on a rack in any one minute is proportional to the sum of the probability that any of the SKUs stowed on that rack are demanded in that minute. The assumption that these probabilities are stable in the interval of analysis implies that $E[B_{k,t}]$ is a constant.

$$b_k \equiv \overline{B_{k,t}} = \sum_{v \in \{i \in \text{rack } k\}} p_i$$

Over the interval of analysis and under the assumption that the distribution of the $p_i$s is skewed heavily to the left (meaning that the bulk of the outbound volume comes from a
single orders for many unique SKUs, which is corroborated by the data), then to a first approximation the function \( A_k(t) \) that describes the accumulation of potential picks on rack \( k \) is:

\[
A_k(t) = b_k t
\]

What is the same:

\[
A_k(t) = \left( \sum_{\nu = [i \in \text{rack } k]} p_i \right) t
\]

Hence the expected potential pile on of a rack is directly related to the product of:

1) The sum of the velocities of the SKUs on that rack expressed as binomial random variables of one trial.

2) The amount of time the rack is allowed to sit before being brought to a station.

Consider now a simplified inventory field where there is one pick station, one bot and \( d \) racks that have \( b_1 = b_2 = \ldots = b_d = b \). Given a desired amount of pile on and a mean mission time for the bot (which will be represented by \( \bar{m} \)), how large should \( d \) be? Clearly \( b \ast \bar{m} \) is equal to the number of potential picks that accumulate on a rack during one mean mission. If we divide desired pile on (represented by \( L \)) by the number of picks accumulated in mean mission time:

\[
\frac{L}{b \ast \bar{m}}
\]

it becomes the number of missions that must take place between when a rack is returned to storage to when it is picked up again (to travel to a station) for it to reach desired pile-on. Hence we need at least enough racks to ensure that the bot is performing a mission at all times and that the rack it's carrying will reach desired pile on by the time it reaches the pick station, hence:

\[
\frac{L}{b \ast \bar{m}} \approx \text{Racks}
\]
Now consider an inventory field with many bots and the $b_k$s distributed as independent Poisson Binomial variables. Then, assuming $Racks \gg Bots$:

$$\frac{L}{b + m} \approx \frac{Racks}{Bots}$$
4. Simulation

The purpose of this chapter is to justify the use of simulation to investigate velocity organization's impact of pile-on, review the literature on velocity organization, parameterize the simulation in terms of the dimensions discussed in Section 3.4.1, and then explain the simulation data structures and algorithm in detail.

4.1 Choice of Methods and Literature

4.1.1 Choice of Methods

Before diving into the specifics of methodology, it's worthwhile to justify the selection of simulation as a way to test the hypothesis that SKU organization on racks influences mean pile-on.

There are two primary options to test the hypothesis: direct experiment and simulation. In fact, the author and colleagues at Amazon did perform a series of small scale experiments with a limited number of racks (<50) and high velocity SKUs (<500). The racks for the experiment were allowed to empty out to a very low gross utilization, and then were filled with high velocity SKUs and allowed to be called to pick stations, but not stow stations. To be more specific, fast selling SKUs were stowed into these racks over the period of four hours and then were restricted from being stowed into for three days. The research team hoped to understand if pile-on would increase for these racks, by how much, and the length of time the effect would be sustained. The data from these experiments is not shown to protect proprietary information, but the author can confirm that pile-on did go up (by close to a factor of 2) for the first 24 hours and remained above baseline for the entirety of the experiment (72 hours).

This approach is limited though. One could reasonably assert that it makes perfect sense that racks stacked with high velocity SKUs would have higher pile-on, but what is the impact on overall system pile-on? Are we just spreading the distribution out, but degrading the mean? Faced with these questions, a researcher has two options: 1) broaden the experiment until a majority of the inventory field is impacted by velocity organization (>1000 racks) or 2) simulate the broader system. A good approach would do both, but in reverse order. Option 1 is expensive: organization of SKUs by humans is costly and for
uncertain benefit⁶. Option 2 is cheap and it allows us to understand if moving forward with option 1 has a higher probability to be worth the expense. Hence, the author selected simulation.

4.1.2 Literature
The method of velocity organization of SKUs (and the use of simulation to test different types of SKU organization) has a long history in the operations research literature. In the early 1960s, Neal [11] proposed velocity organization as a way to reduce picking costs. Heskett [12] [13] followed up with proposing the cube-per-order rule as a metric to organize SKUs. Cube-per-order is a heuristic that organizes inventory by taking the cube (gross volume a SKU occupies) of a SKU, dividing it by the SKU’s average sales per day (velocity), ranking the SKUs in ascending order based on this metric, and applying this ranking to prioritize location of SKUs based on a straight line metric to the I/O area of the pick floor. Heskett’s publication was followed by a bevy of research formalizing his idea and investigating under what conditions it achieved optimality [14] [15] [16] [17] [18] [19] (to name just a few).

In the mid-1970s some of the research community branched out into investigating how warehouse automation was impacted by velocity organization. Hausman [20] in particular introduced the concept of the “crane”: some mechanized object that can travel in one, two or three dimensions to extract a unit of inventory and return it to an I/O location. Hausman and colleagues were the first to apply geometric probability to understand travel distances of cranes. Much research followed on optimizing distance traveled by manipulating pick ordering and the shapes of zones that contain SKU velocity classes [21]. Their work showed that there were often optimal zone sizes and placement given a fixed I/O point of the crane. If the reader is interested in pursuing this topic further, the author can recommend the literature review by Koster and colleagues [22].

---

⁶ Automation would make it less costly. But, for experimental purposes, capital purchases do not make sense.
Much, if not all, of the current research does not apply in a MAR environment because it assumes fixed storage space. As far as the author is aware, this thesis is the first publication to tackle inventory management in the context of MAR automation.

1.4 Inputs

The structure of the simulation to be presented in this chapter depends on the three factors discussed in section 3.4.1 (which are properly represented by distributions) and a scalar (see Table 4.1). The aim of this section is to explore the data around these parameters to understand the constraints of the system and justify choices made in constructing the simulation. It is only after this discussion that we will attack synthesis of the algorithm.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Variable Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binomial Velocities</td>
<td>Distribution</td>
</tr>
<tr>
<td>SKU Spread</td>
<td>Distribution</td>
</tr>
<tr>
<td>Rack Diversity</td>
<td>Distribution</td>
</tr>
<tr>
<td>Mission Time</td>
<td>Expected Value</td>
</tr>
</tbody>
</table>

*Table 4.1: Simulation inputs*

The results presented in section 4.3 were calculated with empirical distributions drawn from Amazon databases. To do optimization in chapter 5, we will abstract these distributions by fitting them to convenient analytical distributions and model empirical distributions. In the sections that follow, the data will be presented followed by the choice of analytical or constructed distribution.

4.2.1 Binomial Velocities

Our first input is the distribution of the velocities of all of the SKUs in the inventory field at some time. In reality, this distribution depends on time in many complicated ways. First, aggregate demand can change with time, meaning that demand for all SKUs can move up or down in tandem. Specific demand for individual SKUs are also dependent on time (especially in the case of a discounting event). Conversely, a SKU can become unpopular and its sales will slack. There is also the complication of the shifting profile of the inventory field as some SKUs stock out and new SKUs are added.
To handle these complications we introduce a simplification: we will only attempt to simulate a relatively small portion of time (on the order of an hour). In an hour we may comfortably ignore aggregate demand changes, specific SKU changes and the influx or outflow of new SKUs as these effects are more potent over longer time scales.

Figure 4.1 shows a histogram of sales per SKU in a typical Amazon robotic warehouse.

![Sales History Histogram](image)

This figure was constructed by summing up all of the shipments out of the warehouse of each SKU at some time \( t \). The sum was over the 30 days that preceded \( t \). To be clear, this figure would not capture the velocity of SKUs that had stocked out at the time that the data was gathered, but the author performed an analysis (not shown) on a different dataset that estimated the strength of this effect and found that it was small. It is clear from Figure 4.1 that Amazon's inventory velocity obeys a Pareto distribution (the reader will note the logarithmic scales of the ordinate and abscissa). The Pareto distribution is a widely observed distribution that is best characterized by the common 80/20 rule. (ex. 20% of SKUS account for 80% of outbound volume). More technically, it is a power law
distribution, where as one moves down the abscissa of a probability density function, the probability density attenuates exponentially.

To calculate binomial velocities, a table was built that included which SKUs were picked in a contiguous 24 hour period in the warehouse under study (n~100,000). This data was then aggregated by summing up the total number of picks for each SKU. This is an important distinction, had we summed up the total number of units picked, it would bias the simulation to compute higher pile-on. Because we are considering only unique SKU pile-on in the simulation, it was necessary to sum the number of times a rack was brought to a station to extract that SKU, regardless of quantity. Once that information was aggregated we merely divide by the number of time intervals present in the 24 hour period (10s in our case, which works out to 8640). This calculation implicitly assumes that aggregate demand is stable over the course of a day, which is incorrect. But the deviations are mild and should not impact the outcomes. We visualize this dataset in Figure 4.2.

![Log Log Velocity Distribution Plot](image)

Figure 4.2: Log Log plot of binomial velocities with the number SKUs that possess that velocity. Data drawn from a 24 hour period of pick data. N~150,000.
Figure 4.2 emphasizes the power law distribution of binomial velocities. A vast majority of sales come from a very select group of SKUs.

The distribution shown in Figure 4.2 was used to do the verification calculation shown in Section 4.3. To perform optimization we will use the fictitious distribution shown in Figure 4.3 and enumerated in Table 4.2.

![Log Log Velocity Distribution Plot](image)

**Figure 4.3: Fictitious binomial velocity distribution.**

<table>
<thead>
<tr>
<th>Binomial Velocity (10s Intervals)</th>
<th>Unique SKUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.85802 x 10^{-6}</td>
<td>900 000</td>
</tr>
<tr>
<td>0.000115741</td>
<td>90 000</td>
</tr>
<tr>
<td>0.00115741</td>
<td>9000</td>
</tr>
<tr>
<td>0.00578035</td>
<td>900</td>
</tr>
<tr>
<td>0.0115741</td>
<td>100</td>
</tr>
</tbody>
</table>

**Table 4.2: Fictitious binomial velocity distribution.**

### 4.2.2 SKU Spread

The next input to the simulation is the relative "spread" of SKUs throughout the field. More precisely, we need a way to determine on how many racks a SKU will be present. A table was built by taking granular bin data from a typical Amazon warehouse, joining it with...
shipment data from the warehouse over a period of fourteen days, and then aggregating the number of racks a SKU was present on each day. The data is summarized in Figures 4.4 and 4.5 (pg. 50).

![Figure 4.4: Combined SKU spread.](image)

Figures 4.4 and 4.5 indicate two items of interest. First, the number of racks a SKU is spread over is Pareto distributed. Second, that distribution depends on the velocity of that SKU. This makes sense because any inventory policy would be based on demand and a forecast for an individual SKU, which would lead to holding higher safety stock for high velocity SKUs. Higher safety stock implies more opportunities for units to be sorted into racks and hence a higher probability that more racks will contain that SKU. This comment explains the long tail for the >10 sales/day case above: a majority of these SKUs would have high forecasts, hence they are represented on more racks. One could also reasonably ask why is the tail for low velocity (<1 sales per day) SKUs long. The answer to this question is that purchasing decisions are made based upon forecasted data, not past sales. It is reasonable to assume that the long tail is accounted for by negative misses in sales.
Figure 4.5: SKU spread based on velocity.
SKU spread is the variable over which managers have the most control with velocity stow strategies. However, we will not explore fitted distributions for this variable because we seek to manipulate the spread deterministically, not statistically. The spread distributions play into the simulation by determining how many racks will contain a specific SKU, but they say nothing about how the SKUs are organized.

4.2.3 Rack Diversity

The final distribution input of the simulation to consider is the number of unique SKUs contained on each rack. A histogram of the data is displayed in Figure 4.6. This distribution was built from granular bin data that was joined to a table that included the bins that were included on individual racks. The SKUs were then organized, aggregated and counted by rack.

![Distribution of SKUs per Rack](image)

Figure 4.6: SKU diversity on Racks.

The shape of the distribution in Figure 4.6 is surprising. One might expect, under the assumptions of random stow (uniform and independent distribution into racks), a sharp distribution with a very small variance. In fact, simple simulations\(^7\) show that we can obtain a normal distribution under the assumptions of random stow, but with roughly half of the standard deviation of the actual data. This unexplained variance is likely a result of

\(^7\) In each time period, 1 unique SKU is placed into one rack from a population of 500. Simulate on 50,000 time periods. Equal probability of rack selection.
bin sizes – racks have various configurations of bin sizes to account for smaller and larger SKUs, the racks that have smaller bin sizes tend to have more SKUs. The fact that we roughly recover the distribution shape from a rudimentary simulation lends some credence to our assumptions about random stow, with caveats for SKU and bin sizes.

The distribution shown in Figure 4.6 is most precisely modeled with a skew normal distribution, but for ease of use an interpretation in the simulation we will use a standard normal distribution (see Figure 4.7).

![Distribution of SKUs per Rack](image)

*Figure 4.7: Probability density of SKU diversity on racks. Actual vs fitted distribution.*

By adopting a standard normal distribution to model SKU diversity we will weigh the lower tail and the upper body of the distribution more heavily than in reality, which is to say that the part of the distribution between the median and the upper tail would be weighted more heavily. This sacrifice is more than compensated for by the simple interpretation of the parameters of a normal distribution.

4.2.4 Mission Time

Using a scalar to represent mission times in our model is easily the greatest simplification over actual reality. Mission times are in fact, widely variable (see Figure 4.8, pg. 53). Our choice is justified for several reasons.
First, it dramatically simplifies the simulation. The current model must track the velocity of every SKU, which rack each SKU is contained on, and every rack’s complete SKU complement. Drawing mission times from a distribution, rather than taking an expected value, increases the complexity of the model by requiring us to track each bot and its associated racks throughout the interval of analysis. This would require implementing a difficult rack indexing system in the code.

Second, it’s not clear that this addition of distribution draw would add much to the results without implementing a full blown spatial model of the inventory field. Our concern is pile-on, not mission times. The main number we are attempting to simulate is how many unique SKU picks are extracted with each mission. We are not concerned with which stations are achieving higher pile-on than the others or how mission times are impacted by various stow strategies.

To conclude: we decided to simplify mission times down to an expected value, rather than a distribution, because to do otherwise would add much complexity for little gain. Instead the simulation works by normalizing mission times to the median mission time and using
mission time to set the length of interval on which demand is simulated with binomial velocities. This simplification is illustrated in Figure 4.9.

A further note here is necessary. The simulation always selects racks based on maximizing pile-on, which means selecting the minimal population of racks necessary to fulfill demand. In reality the process is more complicated, but we cannot provide further details here without revealing sensitive IP. The model accounts for these complexities in a way that minimizes the bias caused by this simplification.

4.3 Data Structures and Algorithm

The simulation works in two broad steps. First, the data structures that represent the inventory field are setup. This step is where the distributions that were discussed in section 4.1 are utilized. The finalized data structure consists of a list of lists. The first level of the list represents the racks in the field, each of the sub-lists contain the SKUs that are

---

8 The median was selected rather than the mean because of the highly skewed nature of the actual distribution. There are small, but significant, outliers far to the right that pull the mean well above the median.
contained on the rack represented by that list. A further ancillary list is created that contains all of the SKUs in the field and their binomial velocities.

Second, the simulation step uses the two lists created in the first step to simulate demand and then calculate the number of racks necessary to bring to station to satisfy that demand and the pile on of those trips. If the number of racks exceeds the number of bots in the field, that demand is carried over into the next interval.

This space has been intentionally left blank
Each SKU is assigned a binomial velocity based on the velocity distribution that describes SKU velocity.

The list passes through an algorithm that assigns a number of representations present in the field based on a collection of distributions. This step outputs a randomized list with each SKU being represented proportional to a random variable drawn from a spread distribution assigned based on velocity.

SKUS are grouped into lists, the length of which are determined by drawing from the diversity distribution.
A binomial draw of one trial (based on the velocity expressed as a p value) is used to determine if there is demand for sku[i] in this interval.

That list is then compared to the list of SKU collections on all racks and a new list is created showing all of the SKUs located on each rack for which there is demand in this interval.

The ranks are then rank ordered by how much pile on they provide. Meaning the rack with the maximum number of SKUs demanded is first, followed by the rack with the second most number of SKUs demanded and so on.

The first rack's pile on is counted. The demand for the SKUs on that rack is deducted from the demand list and the remaining racks. The racks are rank ordered again and the process iterates.

The output is a list containing the number of SKUs for which demand was fulfilled off of that rack. The Length of the list is equivalent to the number of missions required to fulfill the demand.

The process is repeated for the next cycle of demand.
4.4 Simulation Results

It is worthwhile to stop here and question whether the simulation reasonably approximates reality. The algorithm shown in Section 4.2 was built to take as inputs empirical distributions and calculate the consequences to pile-on from SKU rearrangement in the inventory field. The data used to build those distributions is proprietary and cannot be shared, hence we’ve contrived a fictitious velocity distribution and use a normal approximation for the diversity distribution for optimization. To be sure, these distributions are in alignment with reality, but the results will only be directional.

This section shows how the model performs while utilizing empirical distributions built from actual data. The actual pile-on data shown in Figure 4.12 was calculated from a full day of pick data for one of Amazon’s FCs (n~100,000 picks). Strictly this is representative SKUs that sell on average once per day, but the data supports the assertion that our exponential distribution continues to hold as the time scale extends into months (ref. Figure 4.1). The simulation accounts for this probability by taking all of the SKUs in the day in question and giving each of them a velocity equal to the mean velocity of SKUs that sold less than once per day from the dataset used to construct Figure 4.1. The computed distribution, simulated from empirical distributions and assuming a random stow strategy, is shown as well in Figure 4.12.

![Pile-On Actual Vs Simulated](image)

*Figure 4.12: Actual unique SKU pile-on vs simulated.*
At first, Figure 4.12 is disappointing. The simulation fails to capture both the expected value of the actual distribution and the variance. The expected values differ by 1.5% and the difference is statistically significant. This bias is the result of the complexities with mission time that we mentioned in Section 4.2.4. As for the variance, this can be explained by remembering that we assumed velocities remain stable enough to be modeled as binomial draws. This is clearly not the case, but likely explains why the calculated distribution is sharper than the actual.

Figure 4.12 is less disappointing if we consider the purpose of the model: to evaluate different stow strategies on how they influence pile-on. We foreshadow the results of the next chapter in Figure 4.14 (pg. 60).
In Figure 4.14 our 1.5% error is dwarfed by the 25% difference between simulated bifurcation stow and simulated random stow. The effects we are calculating are an order of magnitude greater than the error in our calculation. Therefore, it seems reasonable that the simulation is well matched to our purpose.\footnote{The simulated velocity stow strategy is a segregation bifurcation strategy based on a selling threshold of 1 sale/day. For more information see Section 5.1 and Section 5.2.}
5. Optimization

5.1 Introduction

In the context of this paper, optimization means selection of a SKU organization strategy for the inventory field that maximizes pile-on. Specifically, this means that we are not interested in strategies that materially change the content of the inventory field. We restrict ourselves from diminishing or changing the mix or quantity of SKUs in inventory. A more subtle constraint is that our process for organizing SKUs must be “simple”. Meaning, we seek to achieve desired SKU organization with minimal increases in costs and complexity to existing processes. We will call the former constraint the inventory constraint and the latter the simplicity constraint.

In practice, the simplicity constraint means not adding much manpower or equipment to the upstream processes that feed the stow process path, which stocks the inventory field. In code, our modifications will change the rack assignment step in Figure 4.10. The author’s judgement, informed by his experiences at Amazon, are the foundation for the selection of which processes are “simple”. We will make an attempt to describe in general how such process paths would work in the real world, but this will not be our primary focus.

5.1.1 The Strategies

This chapter will evaluate three velocity stow processes.

1) Diversity – A diversity strategy increases the average number of unique SKUs on each rack. This might seem contrary to the inventory constraint, but it is not necessarily so. If we are able to divide up the units of each SKU and distribute them more broadly across the racks, we may raise the total number of unique SKUs on each rack without increasing the total number of SKUs.

2) Segregation Bifurcation – A segregation bifurcation strategy involves sorting SKUs by some threshold velocity (ex, 1 sale/day, 10 sales/day, etc.). And storing those SKUs that pass the threshold (termed high velocity or hot) in their own rack.
subpopulation and those that do not (termed low velocity or cold) in the same. Once sorted, SKUs are stored randomly in their respective subpopulation.

3) Scatter Bifurcation – A scatter bifurcation strategy also performs a bifurcation but with the aim of spreading high velocity SKUs across all of the racks rather than concentrating them into their own subpopulation.

5.1.2 Baseline

Throughout this discussion we will refer to a random stow strategy as the baseline. A random stow strategy stows SKUs into the rack population independently and uniformly. All of the calculations in this chapter are based on inputs of a normally distributed spread distribution with mean of 150 and standard deviation of 30, the inventory velocity distribution shown in Table 4.2, and the empirical spread distributions developed in Section 4.2.2.

In the course of this chapter, simple diagrams like Figure 5.1 will be used to illustrate the different stow strategies.

![Diagram of random stow strategy]

RU = Random and Uniform

Figure 5.1: Illustration of random stow strategy.

A histogram of the pile on for a random stow strategy is shown in Figure 5.2 and summary statistics are shown in Table 5.1.
Hence, our baseline unique SKU pile on is 1.39. To give an idea of how our output changes with the introduction of the fictitious distributions see Figure 5.3 and Figure 5.4.
We see from Figure 5.3 and Figure 5.4 that the fictitious distribution simulates a higher pile on than the actual data. This is not problematic for three reasons. First, the deviation is easily explained by the fact that the mean of our fictitious velocity distribution is ~10% higher than the mean of the actual velocity distribution. It is clear than a gross increase in overall velocity will push pile-on up. Indeed, the increase in pile-on is on the same order as the increase in the velocity distribution mean. Second, we have already shown that the simulation is a reasonable match to reality when populated with empirical distributions constructed from actual data (ref. Section 4.4). Third, our interest in this section is the relative performance of different stow strategies against the baseline. As long as we use the same distributions to populate the simulations, our results should give good directional guidance.

5.2 Diversity

5.2.1 Mechanism

A diversity strategy attempts to increase the total number of unique SKUs on all of the racks by dividing up the units available before they reach the stow process path. This could be accomplished by introducing a new step between receiving and stow that intentionally mixed the inventory across the various stow stations. This would involve taking large aggregations of units, breaking them up into smaller quantities (i.e. using coke cans rather
than six-packs), and evenly distributing them to the stations. This mechanism is illustrated in Figure 5.5 and Figure 5.6.

Figure 5.5: Diversity uninfluenced by a spreading mechanism.

Figure 5.6: Diversity influenced by a spreading mechanism.
The logic behind this strategy is that with more unique SKUs on each rack, there is a higher likelihood that when the MAR is searching for a match to create pile-on to fulfil demand, a higher mean SKU diversity will create a higher probability that the match will be available to make.

5.2.2 Predictions

In order to examine how increasing rack diversity impacted pile-on, simulations were run with the standard input distributions, but the mean of the normal distribution describing rack diversity was incrementally increased by five. Six runs were performed with means of 150, 155, 160, 165, 170 and 175 SKUs respectively. The results are shown in Figure 5.7 and Figure 5.8.

![Increasing Diversity impact on Pile-on](image)

*Figure 5.7: Impact on pile-on of increasing mean diversity.*
Our analysis indicates that by increasing mean diversity there is no significant impact on mean pile-on. Figure 5.8 surprises, but the result can be explained by the Pareto distribution of SKU velocity. If we return for a second to the conclusions of Section 3.4.3:

$$A_k(t) = \left( \sum_{v \in \{i \in \text{rack } k\}} p_i \right) t$$

And remind ourselves that $A_k(t)$ represents the buildup of potential picks on a rack in some time interval, we can see that the results shown in Figure 5.7 and Figure 5.8 make sense if **SKUs are chosen randomly** to be split into smaller aggregations. This means that it is highly probable that the SKUs chosen for one rack will all be low velocity, adding very little to the buildup sum:

$$\sum_{v \in \{i \in \text{rack } k\}} p_i$$

And hence keeping pile-on stable.
5.3 Segregation

5.3.1 Mechanism

The segregation bifurcation strategy works by a threshold sortation of SKUs. If a SKU is forecast to sell more than some amount per unit of time, it is considered a fast SKU and is sorted into a rack that only holds fast SKUs. If a SKU is forecast to sell less than some amount per unit time it is classified as a slow moving SKU and stored into a rack that only holds slow SKUs.

An obvious criticism of this approach is that forecasts are inherently imprecise tools. There are two interrelated responses to this critique. First, calculating the probability that a SKU will surpass a threshold is a much different task then trying to determine the precise number of units that will sell. It is a simpler prediction and thus has tighter error bounds. Second, pile-on is insensitive to erring to inclusiveness. If a SKU is classified as high velocity, but in fact develops into a slow moving SKU, it does not detract from the speed of the rack that it resides on other than to take up space that could be occupied by other fast moving SKUs. There are no negative build up effects (ref. Section 3.4).

The mechanism for bifurcation is illustrated in Figure 5.9.

![Figure 5.9: Segregation bifurcation illustrated](image)
3.2 Predictions

For this set of simulations, a sorter was built into the code that would separate SKUs as high velocity or low velocity classes and then assign them to racks appropriately. The variable in this simulation is the threshold\(^\text{10}\). Three runs of the simulation were performed, each with a different sales per day threshold set: 1 sale/day, 10 sales/day and 50 sales/day. The histograms of the respective pile-on are shown in Figure 5.10.

\(^{10}\) Another variable could potentially be number of classes, but the author discarded this possibility because he judged that it would violate the simplicity constraint.
Figure 5.10: Segregation pile-on simulation results.
To make things a bit clearer, Figure 5.11 shows the mean pile-on for each threshold with a bootstrapped 95% confidence interval.

![Figure 5.11: Mean pile-on by bifurcation threshold with confidence intervals.](image)

<table>
<thead>
<tr>
<th>Bif. Threshold</th>
<th>Mean Pile-on</th>
<th>Over Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.53</td>
<td>10%</td>
</tr>
<tr>
<td>10</td>
<td>1.60</td>
<td>15%</td>
</tr>
<tr>
<td>50</td>
<td>1.79</td>
<td>29%</td>
</tr>
</tbody>
</table>

Table 5.2: Mean pile-on for different bifurcation thresholds.

We may draw two conclusions from the information presented here. First, a segregation strategy based on a threshold bifurcation is a dramatic improvement over the baseline, it boosts pile-on by as much as 29% in our directional simulation. This conforms to calculations done with full empirical distributions (ref. Section 4.4). Second, the optimum seems to be in the upper range of the threshold, but further investigation is necessary to determine how these ranges impact mission times.

Several further avenues of investigation present themselves. First, it would be interesting to test how combining a diversity strategy within the subpopulation of fast moving racks would impact pile-on. The results from Section 5.2.2 notwithstanding, it seems that expanding diversity within the fast moving racks would drive up pile-on because the selection would necessarily add fast moving SKUs to the racks. Second, a more granular investigation of threshold conditions (i.e. setting the threshold at >20/day, >30 per day, etc.) to understand the shape of the function. Third, it would be useful to recalculate the simulation with a constraint that states the minimum number of racks a fast moving SKU
must be present on (i.e. a SKU with x velocity must be on y number of racks). This would allow us to understand what optimum could be achieved while also taking into account the operational difficulties of concentrating a single SKU on too few racks.

5.4 Scatter

5.4.1 Mechanism

A scatter bifurcation strategy operates by performing a threshold bifurcation as in the segregation bifurcation strategy, but then using that organization to place fast moving SKUs on all of the racks rather than a subpopulation. The scatter strategy is illustrated in Figure 5.12. The logic for this strategy is that by scattering high velocity SKUs over all of the racks that the MAR will have a higher probability of matching demand for a high velocity SKU with a low velocity SKU to create pile-on.

![Diagram of Scatter](image)

**Figure 5.12: Scatter illustrated.**

5.4.2 Predictions

The same sorter that was employed to perform the bifurcation simulation, and the same thresholds, were used in these simulations. The difference was that the high velocity SKUs were assured to be normally distributed over each rack. The simulation results are shown in Figure 5.13.
Figure 5.13: Scatter pile-on simulation results.
We see from Figure 5.13 that scattering high velocity SKUs has very little impact on pile-on. In fact, it’s not clear that the mean moves at all (see Figure 5.14).

![Figure 5.14: Bootstrapped 95% mean confidence intervals for scatter simulations.](image)

<table>
<thead>
<tr>
<th>Bif. Threshold</th>
<th>Mean Pile-on</th>
<th>Over Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.43</td>
<td>2.7%</td>
</tr>
<tr>
<td>10</td>
<td>1.44</td>
<td>3.3%</td>
</tr>
<tr>
<td>50</td>
<td>1.44</td>
<td>3.6%</td>
</tr>
</tbody>
</table>

Table 5.3: Scatter simulations mean pile-on

5.5 Conclusions

The results from the simulations presented in this chapter indicate three things.

1) Raising the number of SKUs on a rack, if drawn randomly, does not increase pile-on.
2) Segregation bifurcation strategies have a large positive influence on pile-on. Further investigation is necessary to ensure that this strategy maintains other important system parameters (i.e. mission times, number of missions) within acceptable bounds.
3) Scatter bifurcation does not have a meaningful impact on pile-on.

We return for a moment to our objective:

*Maximize pick rate by minimizing cycle time and maximizing operator utilization without increasing the number of bots in the field.*

With the completion of this chapter and the listing of its conclusions, we have resolved the operator utilization portion of our objective. We may suggest that a segregation bifurcation strategy will likely alleviate the constraints on cycle time. We now move on to the issue of cycle time itself.
6. Cycle Time

6.1 Introduction

The next two chapters concern themselves with minimizing cycle time, in accord with our objective:

*Maximize pick rate by minimizing cycle time and maximizing operator utilization without increasing the number of bots in the field.*

This chapter will focus on explaining the variation we see in Figure 6.1, which shows the daily mean of cycle times for the pickers working in four different Amazon fulfillment centers.

![Daily Mean Cycle Time Variation](image)

*Figure 6.1: Mean cycle time variation over a six month period for four Amazon FCs (excludes peak). Colors represent distinct FCs and are consistent throughout figures.*

We may make three observations about this data:

1) There is a weekly fluctuation around some expected value.
2) The expected value is not stable in time.
3) The expected value varies between the various buildings.

Figure 6.2 shows the distribution of cycle times over the course of a day at an Amazon FC. It emphasizes the variability of the distribution at a more granular level. Figure 6.3 shows the variation between mean cycle times at the worker level over the course of a month.
The first question one may reasonably ask is if the variations we observe have a meaningful impact on operations. Figure 6.1 shows some of the buildings had mean fluctuations on the order of 2 seconds. In the neighborhood of cycle times we are exploring, a mean increase
of 2 seconds translates into an 18% loss of throughput. Mean cycle time is the single most important driver of throughput at the inventory field.

Upon initial examination, the variation is puzzling. MARs are standardized systems, designed with the intent that stations be essentially interchangeable. Why, therefore, is there such variation in time and across buildings?

Our approach involves application of statistical models to search for parameters that accurately predict cycle times. With these parameters in hand, we will then discuss ways to use them to influence cycle times.

6.2 Structure
This chapter is structured differently than the previous five. Rather than presenting a straightforward attack on the problem, we will first go down a rabbit hole. We will do this for two reasons:

First, the “rabbit hole” (termed the mechanistic model) will illustrate a range of factors that have virtually no impact on cycle times. Hence, it excludes a lot of dimensions that one might consider in attempting to understand cycle times. Sometimes knowing that a parameter is not influential is as valuable as knowing it is.

Second, our review of the mechanistic model will illustrate a way of thinking that hindered the author in his attempt to understand the problem. A discussion that will be useful later on.

6.3 The Mechanistic Model
6.3.1 Approach
One of the fundamental purposes of automation is to remove human variability from production. In that spirit, when the author set out to look for factors that explained variation in cycle time, he searched for those that could be machine controlled.

Picking is not a complicated process. It is worth reviewing Figure 2.3 here without the clutter of nomenclature.
A picker's job consists of 1) reviewing the computer screen next to them to identify the SKU to be picked and where it is located on the rack, 2) reaching into the bin to grab the item, 3) placing the item in a tote to pass it further down the line.

Of these steps, only step 2 involves any kind of variability. The bins themselves are a huge mixture of SKUs of different sizes, quantities and organization. Hence, it seemed reasonable that this variability accounted for the variability in cycle times. Moreover, metrics like these could be controlled by imposing stow controls on bins. This would work by having the system track certain metrics about bins and prevent stowers from placing inventory into bins that exceeded threshold conditions.

6.3.2 Data

To test the hypothesis of bin variability as the source of cycle time variability, a dataset was constructed that consisted of multiple regressors – see Table 6.1. The construction of the dataset consisted of combining pick data with inventory data that described the state of the bins at the time of the pick.
<table>
<thead>
<tr>
<th>Response</th>
<th>Definition</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cycle time</td>
<td>The amount of time a pick takes</td>
<td>MS</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Regressor</th>
<th>Definition</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records</td>
<td>Unique SKUs in bin</td>
<td>Integer</td>
</tr>
<tr>
<td>Units</td>
<td>Number of units in bin</td>
<td>Integer</td>
</tr>
<tr>
<td>Utilization</td>
<td>Total cube of units in bin divided by bin capacity.</td>
<td>Continuous between 0 and 1</td>
</tr>
<tr>
<td>Mass</td>
<td>Mass of product picked</td>
<td>Grams</td>
</tr>
<tr>
<td>Volume</td>
<td>Cube of product moved</td>
<td>CM^3</td>
</tr>
<tr>
<td>Time</td>
<td>Time pick occurred</td>
<td>Continuous between 0 and 24</td>
</tr>
<tr>
<td>Hours</td>
<td>Number of hours a picker has worked at a pick station over career.</td>
<td>Hours</td>
</tr>
<tr>
<td>Template</td>
<td>Bin template</td>
<td>Categorical</td>
</tr>
<tr>
<td>Quantity</td>
<td>Number of same SKUs picked from bin in a quantity pick</td>
<td>Integer</td>
</tr>
<tr>
<td>Quantity Pick</td>
<td>Was the pick a quantity pick?</td>
<td>Categorical</td>
</tr>
<tr>
<td>Height</td>
<td>Height of bin above ground</td>
<td>MM</td>
</tr>
</tbody>
</table>

Table 6.1: Regressors and response descriptions for cycle time modeling.

Figure 6.5: Distributions of various parameters from cycle time dataset
The total dataset had on the order of 100,000 unique picks over the course of a day. The shape of some of these distributions can be seen in Figure 6.5.

6.3.3 Results

The results of model construction on this dataset were abysmal. The best model the author could construct using linear regression (with nonlinear variables) had the following diagnostics.

![Residuals Vs Response Plot](image)

*Figure 6.6: Cycle time model response vs residual plot. Artifact is the result of outlier removal method employed.*

![Standardized Residual Plot](image)

*Figure 6.7: Standardized residual plot.*
Further, the model shown had an adjusted $R^2$ of .27. Disappointed with that result, the author turned to different techniques and model types. First, a variety of families of nonlinear models were fitted to the data, then generalized linear models using different error distributions were tried, finally machine learning techniques were applied. None of those techniques yielded a fit better than the linear model presented in this section.

6.4.1 Employee Experience

The only variable that explains aggregate cycle times meaningfully is picker experience. After struggling to understand granular cycle time data, the author constructed new datasets that consisted of cycle times that were averaged over weeks and months for all of the pickers working in the MAR facilities in the Amazon network for six calendar months. When plotted against the amount of experience the pickers had accumulated in the preceding time period, learning curve patterns developed (see Figure 6.8).
Figure 6.8: Monthly mean cycle times plotted against picker experience in hours. Each dot in the above plots represents one picker in Amazon MAR network. Vertical solid line represents training period cutoff. Solid horizontal line represents average value necessary to achieve throughput targets. Months are calendar months – does not include peak.

Figure 6.8 deserves some explanation. Each month in the figure is in chronological order. The vertical axis plots monthly mean cycle time for an individual picker. The horizontal axis plots the hours of experience picking (i.e. logged into a pick station) that that picker had before the month began. The horizontal solid line represents the mean cycle time necessary to achieve desired throughput targets. The vertical solid line represents the end
of the Amazon training period for new pickers. By taking the median of pickers with roughly equivalent amount of experience we may plot Figure 6.9.

We may make several observations from Figure 6.8 and Figure 6.9. First, it seems that as pickers become more experienced, on average they become faster up to a point where they settle down to a consistent aggregate cycle time. Second, this settling happens well below
the mean cycle time necessary to achieve throughput targets. Third, the settling generally happens in twice the amount of time Amazon reserves for training.

These observations lead us to a new hypothesis: might the fraction of inexperienced pickers in the pool account for aggregate changes in cycle time? Figure 6.10 and Figure 6.11 are highly suggestive.

**Figure 6.10:** Median network cycle times for each month. Dashed lines represent bootstrapped 95% confidence intervals.

**Figure 6.11:** Vertical axis is the fraction of man-hours worked by pickers who were not out of the Amazon training period for that month. Horizontal axis is the month. Vertical axis ranges over ~20%.
It seems likely that there is a connection between inexperienced hours worked, defined as the fraction of total man-hours worked by pickers in Amazon MAR facilities not out of the training period, and global median cycle times.

6.4.2 Model
To test the hypothesis of inexperienced workers influencing aggregate cycle times a more granular dataset was constructed. This dataset is similar in every respect to the monthly dataset used to construct Figures 6.7-6.10, except that it was built on a weekly basis over the same time period\textsuperscript{11,12}. This gives us an $n$ of 26. After construction, the autocorrelation and partial-autocorrelation functions were examined and six lags were found to be influential, hence models of the following form were fitted to the data:

$$\tilde{C}_t = a + b \, I + c \, \tilde{C}_{t-1} + \varepsilon$$

Where $\tilde{C}_t$ is the median cycle time in week $t$ and $I$ is the fraction hours worked by inexperienced pickers in time period $t$. Inexperienced is defined as workers not out of the Amazon defined learning curve. In words, the cycle time is determined by the cycle time in the last period, plus some factor proportional to the fraction of inexperienced hours worked.

6.4.3 Diagnostics
The simple model proposed in the above sections predicts the data remarkably well with reasonably good diagnostics (Figure 6.12). A few caveats are in order. There is clearly some time dependent structure in our diagnostics, but this results from the fact that we have only included one autoregressive term in the model where good practice would dictate six. This decision is justified by the low number of data points that we have available and in keeping with the spirit of Occam’s razor. In the end, we have explained close to 90% of the variability of median cycle times.

\textsuperscript{11} This dataset includes all pickers from Amazon robotic network.
\textsuperscript{12} The results of this test were consistent with the monthly dataset.
6.4.4 Anecdotal Evidence

The above arguments do a good job explaining observation (2) from Section 6.1, but what about the other observations? The author cannot offer a quantitative analysis, but may submit some unsupported explanations.

The weekly variation observed in Figure 6.1 could be the result of the fact that certain parts of the week are "weighted" with more experienced workers by dint of the fact that certain shift types are preferable and shift allocation is done by seniority. There is also incentive
by management to stack certain parts of the week with high performers to meet weekly peak volume targets.

As for variation between buildings, see Figure 6.12.

![Inexperience Hours Worked by FC](image)

Figure 6.13: Hours worked by inexperienced pickers as a fraction of total man-hours worked at pick stations. Note: colors correspond to Figure 6.1.

### 6.5 Impacts

The model formulated in the last section allows us to ask the question: what kind of throughput could be achieved if we held the fraction of inexperienced workers to some predetermined level? Before we approach this question though, a few assumptions must be stated. First, we must assume that downstream processes are unconstrained. Hence pick is the bottleneck and downstream processes will be scaled to preserve that status. Second, we must assume that some strategy has been implemented to increase pile-on, such as bifurcation segregation. Without this implementation, we would run into the bot demand constraint, which would retard the benefit of decreasing aggregate cycle times. With those caveats stated, we may apply the model and review the results. See Figure 6.14.
Figure 6.14 was constructed by taking as inputs: a representative median cycle time of the Amazon network and a reduced aggregate fraction of inexperienced hours worked, and calculated the evolution of median cycle times over the following four months.

To understand the shape of the curves, we must think about the forces in play. The scenarios plotted in Figure 6.14 assume not only that the fraction of inexperienced hours worked are reduced, but they are held at a constant level. Hence, the fraction of the workforce that is designated as experienced maintains a constant level as well. But, a significant fraction of the experienced cohort (as defined by our model) is not yet out of the true learning curve (as opposed to the Amazon defined one, which the reader will recall is how we defined “inexperienced”). Hence, if the inexperienced cohort is held steady, the experienced cohort climbs down the curve, until they reach a point where all of the experienced workers are at the bottom. This explains the decline and plateau of median cycle times predicted by the model.

How does this translate into pick rates? See Figure 6.15.
In Figure 6.15 we assumed a constant OU figure and applied our pick rate equation:

\[
\text{Pick Rate} = \frac{3600 \cdot \text{Operator Utilization}}{\text{Cycle Time}}
\]

We can see the inverse quadratic effect that changes in cycle time have on pick rate (ex. a 20% reduction in cycle time translates into a 25% increase). As we have assumed that pick rates are directly proportional to throughput, one could also view Figure 6.15 as a direct plot on the increase in throughput related to reductions in the fraction of inexperienced hours worked.

The author is not naïve to the difficulty in reducing inexperienced hours worked by 75% and it would be imprudent to suggest that such a reduction is possible only through the methods discussed in this thesis. But, it does seem reasonable to suggest that a 10% throughput increase is possible with disciplined application of policies to reduce inexperienced hours worked.
6.6 Direction

From the above discussion we can conclude that the fraction of man-hours in a period of time worked by inexperienced pickers contributes significantly to increases in median pick cycle times. Our goal is to hold median cycle times down, hence decreasing the fraction of hours worked by inexperienced pickers seems to be the path forward. This will be the subject of the next chapter.
7. Turnover

7.1 Introduction

The author’s intent in this chapter is to provide a short, practical, non-controversial guide to increasing employee job satisfaction and hence decreasing turnover. Also included is a discussion of the authors own thoughts on retaining employees based on his experiences as both a manager and a line employee.

Turnover and absenteeism is a widely studied phenomena in the fields of organizational psychology and behavior, research has been ongoing for close to 60 years [23]. The goal of this chapter is to review what the determinants are on this topic and to make general suggestions to managers running MAR facilities on how to decrease their turnover. In the spirit of generality that this thesis aims for, this chapter will not discuss Amazon’s staffing or employee policies nor make recommendations specific to Amazon. That direction would overly focus the discussion. In addition, the author was not in a position to gain an objective nor complete view of employee policy at Amazon.

Two further caveats before continuing. First, the topic of turnover is vast. A synthesis of the literature could occupy an entire book. The author has selected topics based on two criteria: A) These influences should be readily manipulated by middle managers, hence we avoid larger policy decisions; B) the topics should relate in some way to line employees. This second point is subtle. Much of the literature surrounding employee retention focuses on “top talent” (i.e. highly educated or skilled professionals). These topics, while important, are not entirely relevant to our discussion. As these two criteria are subjective, the selection is inherently incomplete and subject to the author’s biases. Second, the literature contains many contentious issues [24]. The author has done his best to present what is widely accepted and avoid entering into any debate.
7.2 Influences on Turnover

7.2.1 General Influences

Three broad influences are identified in the literature for employee turnover: personal, work, and environmental. Personal factors include demographics, age, geography, and life events. Work factors are the parts of organizational culture, structure and politics that directly impact employees. Environmental factors are those societal influences that impact normative views of work [23]. Of these three factors, the first is not easily influenced by management after setting the bounds on a work-life balance. The third is also difficult to influence, but must be adapted to. For example, it is well documented that American’s views of what constitutes a “good” job changed in the last half of the 20th century [23]. Katzell notes there are a “[r]evised definitions of success, with less emphasis on material achievement and more on personal fulfillment” [25]. As these normative definitions change, management needs to adapt the jobs they provide to meet these expectations, if they seek to retain workers. This brings us to the second factor listed above: work. Work factors are those that are largely within the purview of management to influence: compensation, work hours, job design, and line-management.

7.2.2 Compensation

Compensation is highly influential on employee retention. One of the most important papers published about this phenomenon is [26]. This study is distinguished by a large dataset (n~5000) that is complete in the sense that all exempt hires for a firm over a period of three years were tracked. Hence there is a wide variety of different functions and rolls represented. Two features that limit the results are 1) the study is confined to one firm 2) that they only represent hires that are salaried. Figure 7.1 is drawn from this paper (pg. 93).
Figure 7.1 is interesting for several reasons. It plots the survival function over a four year period for employees that have been ranked according to their performance. First, we note the convex nature of the plot: essentially that the highest voluntary turnover is observed in low and high performers. Second we note that high salary growth (percent annual pay increases) dramatically impacts the shape and level of the survival function.

There is also evidence that organizations that do not perform some merit based compensation reward system face higher turnover with their top performers [27].

1.2.3 Hours and Flexibility

There is a large body of evidence that indicates that flexible working hours (defined as maximizing employee control over their schedules) have a positive impact on employee happiness and satisfaction, hence decreasing turnover [28]. Increasing flexibility of hours can be difficult in the context of a manufacturing environment. Several manufacturing specific suggestions are offered in [29], of which a particularly intriguing idea is to use vacation time as a reward for task completion, rather than base it solely on hours worked.
7.2.4 Job Design

Job design is a notoriously difficult phrase to define, but it consists largely of the idea that one should match employee's preferences for work with the work they do and consistently giving employees opportunities for growth. This brings us to the difficult subject of intrinsic motivation [30]- the factors that actually motivates employees to do good work. According to the lauded meta-study by Herzberg [31] the factors that cause job satisfaction are not the same as the factors that cause job dissatisfaction (see Figure 7.2)

![Figure 7.2: Exhibit 1 from Herzberg [31].](image)

Herzberg terms this dichotomy issues of motivation and issues of hygiene. The primary sources of motivation for employees are achievement, recognition and "work itself".
whereas hygiene consists of compensation, management and administration. Herzberg uses these findings to argue that designing jobs better (what he calls vertical loading) is the way to achieve higher motivation and hence higher satisfaction. He provides a prescriptive set of suggestions in his paper, which are displayed in Figure 7.3.

<table>
<thead>
<tr>
<th>Exhibit III</th>
<th>Principles of vertical job loading</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle</td>
<td>Motivators involved</td>
</tr>
<tr>
<td>A</td>
<td>Removing some controls while retaining accountability</td>
</tr>
<tr>
<td>B</td>
<td>Increasing the accountability of individuals for own work</td>
</tr>
<tr>
<td>C</td>
<td>Giving a person a complete natural unit of work (module, division, area, and so on)</td>
</tr>
<tr>
<td>D</td>
<td>Granting additional authority to employees in their activity: job freedom</td>
</tr>
<tr>
<td>E</td>
<td>Making periodic reports directly available to the workers themselves rather than to supervisors</td>
</tr>
<tr>
<td>F</td>
<td>Introducing new and more difficult tasks not previously handled</td>
</tr>
<tr>
<td>G</td>
<td>Assigning individuals specific or specialized tasks, enabling them to become experts</td>
</tr>
</tbody>
</table>

*Figure 7.3: Exhibit 3 from Herzberg [31].*

7.2.5 Line-Management

The final topic in our tour of controllable factors that influence turnover is line management itself. Though it is difficult to measure the impacts of single line managers on gross turnover outcomes, it is not controversial to assert that direct management has a high impact on job satisfaction. Creelman and Hunt [32] outline a simple, action-item-based plan to help senior managers understand the impacts of line management on their turnover, mitigate the negative effects, and strengthen the positive.
Their action items are (direct quotations):

1) Reject the idea that turnover of hourly works is unimportant or uncontrollable.
2) Penalize or move managers who generate excessive turnover.
3) Check if company-induced demands are actively undermining managers’ ability to retain staff.
4) Don’t assume that all hourly workers share identical needs and motives.
5) Management must support the initiative to retain employees with their words and deeds.

Another relevant section from Creelman and Hunt’s paper relates how organizations can unintentionally incentivize behavior (in their line managers) that increases turnover. They suggest various ways to avoid this:

1) Don’t give line managers too wide a span of control. A direct quote: “A manager with too many direct reports cannot give the direction, feedback, and support needed to get people to stay”.
2) Do not pressure line managers to the point where they emphasize short term targets over long term growth and health of their team.
3) Do not give line managers too much administrative work. The organization should attempt to minimize administration work for line managers as much as possible.
4) Attempt to give line managers the best leadership possible. This item invokes the principle of modeling desirable behavior.

7.3 Discussion

One could reasonably ask what all of this means for warehousing. In this section the author will attempt to color the current topic with his own observations about how this relates to automated operations. For the duration of this section the author will make assertions on no other authority than his own experience. The reader is duly cautioned.

One of the greatest difficulties in warehousing in the context of higher automation is the drudgery of the work. This is by design. Automation tends to have the effect of limiting the amount of variability that operators are exposed to. This results in tasks that are monotonous. We face a countervailing difficulty in the fact that as workers gain
experience, they tend to get better at their task. This is a very old problem. In fact, it has existed since the beginning of assembly line techniques. Henry Ford, famous for raising wages to $5 a day, embarked on this change because he faced massive attrition in his factories that destroyed productivity. Those workers preferred to find employment in small shops where they could work on a variety of tasks, rather than perform one task over and over [33].

What's to be done about this? It is not a norm of business that work ought to be enjoyable, but why? Faced with a tradeoff between productivity and enjoyment, it's clear that productivity should prevail, but it's not clear that such a tradeoff exists. With increasing automation, workers become more and more reliant on computerized systems to perform their daily tasks. There are opportunities to exploit these systems through gamification, personal development, and passive environmental stimulus (ex. music) that could vastly increase the enjoyment of work while causing minimal increases in expense. To be clear, the author is not advocating that we introduce these changes without consideration to their impact on productivity or safety, only that the connections are largely untested. We should test them. The author firmly believes that if work could be made more enjoyable, workers would have higher motivation in the sense of Herzberg [31], which would be beneficial to both workers and businesses.

The suggestions made by the literature as it relates to hours and pay are telling. The literature abounds with estimates of the cost to replace a worker and how the efficiency of businesses suffer under high turnover conditions. These estimates vary widely, but all agree that there is some cost and it is significant. A potentially good policy would be to make a careful estimate of the cost per capita of turnover and use that figure to calculate an optimum turnover ratio. This figure could then be used to drive wage decisions. If the author had to guess, many companies in the warehousing industry are incurring greater costs than they are saving with their wage rates because of the loss of human learning in their systems (along with the attendant costs of finding new people and onboarding them).

The author believes that schedule flexibility, if done carefully and prudently, is a no-cost way of increasing employee retention, regardless of the type of business.
Finally we come to line management. As someone who has worked as a manual laborer, full time, for low wages, the author believes this issue trumps all others in importance. Many jobs that do not require a high degree of skill are substantially the same in content and earning ability. Good management easily makes the difference between a good job and bad job. Building a competent line management corps is an incredibly difficult task. The author finds much to like in the suggestions offered by Creelman and Hunt [32], but acknowledges that there are no easy paths.

A final note: the author suggests skepticism in any automation that purports to mitigate the impact of human learning on processes, unless that automation directly performs the task under analysis. One of the most important conclusions stemming from this thesis is experience is not just the most important predictor of median cycle times, but the only meaningful one, as evidenced by the failure of the mechanistic models presented in Section 6.3.

7.4 Summary

This chapter has provided a short introduction to the academic conclusions around managing employees to reduce turnover and the author's thoughts on how this relates to the topic at hand. Hopefully what has been provided can increase awareness around factors that make employees quit and provide practical ways to increase employee retention.
8. Conclusion

This chapter provides a short summary section of what has been demonstrated in this thesis, a discussion of costs and benefits, a review of the generality of results, and a speculative section that discusses broadly some implications of the work here.

8.1 Summary

This thesis has accomplished three things:

1) It stated an objective function of maximizing throughput thorough a MAR facility by driving pick rate, which is a function of both the efficiency of the bots bringing inventory to the operators and how fast the operators worked. This optimization will hold given that stations downstream of the pick process path are unconstrained (i.e. not bottlenecked).

2) It argued that the efficiency of bots could be increased by performing a velocity stow strategy called segregation bifurcation. Segregation bifurcation means splitting inventory into two pools of high selling and low selling inventory based on a threshold condition (ex: 1 sale/day) and stowing them into distinct subpopulations of racks. Simulations suggest that the higher the threshold, the greater the positive impact on pile-on, but one should note that the smaller the subpopulation of racks the more likely those racks will bottleneck the system.

3) It demonstrated that the single most influential predictor of mean cycle time was the number of inexperienced hours worked in the pick process path. This suggested that a way to maximize pick rate would be to reduce employee turnover and hence decrease the number of inexperienced hours worked. The thesis attempted to give several practical suggestions for how to influence employee turnover.

8.2 Costs and Savings

In this section we will estimate the financial benefit of applying the two results of this thesis. We will examine the variable cost structure changes with a toy financial model. Before we delve into this discussion though, it's worth talking about the costs of implementing the two suggested changes.
Velocity bifurcation potentially can be done completely through capex. The author cannot
go into details, but his experiences at Amazon have convinced him that it can be done with
(at most) one additional touch on a fraction of inbound volume. Most of the financial pain
comes in the form of lost efficiency from process changes and fixed cost investment.
Therefore, we will ignore these costs in the variable costs estimates.

Reducing turnover could involve additional variable cost in the form of higher wages, more
benefits, or additional management. It could also come in the form of investment in policy
changes, work improvement, training or any other number of factors. The combinations
are infinite. Dealing with this complexity is not easy and we will have to resort to a crude
approximation.

It's worth noting that there are capital cost benefits to improving throughput. The most
notable being that a sizable increase in throughput could delay the need to expand capacity
as the operation scales, freeing up cash for other activities. The size of his benefit will
depend on the length of time new investment can be delayed and the discount rate of the
organization. Additionally, increasing pile-on by the methods described in this thesis could
reduce the number of drives necessary to service stations, reducing total capex necessary
to build and maintain a MAR facility.

8.2.1 Variable Costs

To place an estimate on the variable cost benefit, we will build two toy financial models
based on a simplified cost structure of a warehouse. (It should be noted, the numbers that
follow are complete fiction. They have no resemblance to Amazon operations.) In our
warehouse, all staffing decisions are based on pick rate. We have a maximum number of
pick and stow stations and we always fully utilize. We assume that the stowers can stow at
the same rate of the pickers. We have fixed rates for receiving and outbound operations,
and add people to those operations to meet the volume of the pickers. Our network
consists of one MAR facility. Our model is shown in Table 8.1.
Several features of Table 8.1 are notable. We have approximated the additional variable cost of lowering turnover with a fifty cent increase in the wage rate with fringes. How this cost is distributed is dependent on the organization, but here we have assumed that the additional money is paid out to every worker in the warehouse, not just those working at the inventory field. The model is sensitive to the wage rate, as it goes down, the savings become less favorable; as it goes up, they become more favorable. The model is also
sensitive to the mean units per order figure, becoming more favorable as it goes up and less as it goes down. Overall, we have a 2.7% savings per unit for the 10% increase case and a 4.6% savings per unit in the 15% case.

Another interesting consideration is the case where operations consists of a network: one manual facility and one robotic facility. In this case, because of the favorable cost structure of the robotic facility it makes sense to run the robotic facility at full capacity and use the manual facility as a buffer to fulfill any peaks in demand. The model displayed in Table 8.2 deals with this case.

<table>
<thead>
<tr>
<th>Baseline</th>
<th>10%</th>
<th>15%</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Robotic</strong></td>
<td><strong>Manual</strong></td>
<td><strong>Robotic</strong></td>
</tr>
<tr>
<td><strong>Inbound</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Receiving Rate</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>Total Receivers</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>Hourly Mean Pick Rate</td>
<td>225</td>
<td>115</td>
</tr>
<tr>
<td>Total Stowers</td>
<td>50</td>
<td>77</td>
</tr>
<tr>
<td>Total Pickers</td>
<td>50</td>
<td>77</td>
</tr>
<tr>
<td>Weekly Units Throughput</td>
<td>450000</td>
<td>350000</td>
</tr>
<tr>
<td><strong>Outbound</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean Units Per Order</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>Orders</td>
<td>375000</td>
<td>291667</td>
</tr>
<tr>
<td>Rate</td>
<td>125</td>
<td>125</td>
</tr>
<tr>
<td>Outbound Workers</td>
<td>75</td>
<td>59</td>
</tr>
<tr>
<td>Total Hours Worked</td>
<td>7,480</td>
<td>8,880</td>
</tr>
<tr>
<td>Wage Rate with Fringes</td>
<td>$20.00</td>
<td>$20.00</td>
</tr>
<tr>
<td>Weekly Variable Costs</td>
<td>$327,200.00</td>
<td>$315,120.00</td>
</tr>
<tr>
<td>VCPU</td>
<td>$0.409</td>
<td>$0.394</td>
</tr>
<tr>
<td>Improvement Over Baseline</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8.2: Financial model, two warehouses; one robotic, one manual.

The model shown in Table 8.2 is constructed with the same assumptions of Table 8.1 with the added assumptions of a fixed demand of 800,000 units per week. The business
optimizes on costs by using all of the capacity in the robotic warehouse and then shifting over the manual operations to reach peaks. The pick rate is fixed in the manual warehouse and pickers and stowers are adjusted to meet volume targets. We have also assumed that the additional variable costs for reducing turnover in the robotic warehouse only accrue to that warehouse. Our savings are 3.7% per unit for a 10% increase in throughput and 6.35% per unit for a 15% increase.

8.3 Generality of Results

For this section we address the generality of the two results (velocity organization within MARs and learning curves) separately.

Unfortunately, the velocity organization result probably does not have much applicability outside of the narrow confines of MAR automation of warehousing. It is possible to imagine other automation systems that operate by bringing the storage (not just the SKUs) to the pickers, but whether different types of MAR systems emerge is yet to be seen. Happily, our results confirm a general trend within the OR literature that, in general, organizing SKUs by velocity tends to lower costs.

The learning curve result has more generality in that it reinforces what we already know: learning curves are everywhere and highly influential on system performance [34]. This result is especially important in light of MAR facilities level of automation. Even in a technologically driven warehouse, the biggest lever to move volume is human learning.

8.4 Final Thoughts

The subject of automation is one of immense import to operations and manufacturing professionals. Aside from (or because of) the cost and quality aspects of automation, it appeals to the cultural ideals of manufacturing: it reduces overall variability in processes, which leads to higher consistency. One could certainly argue that consistency is the highest ideal of our culture: to deliver on time, at any time, under any condition. Consistency is the product of speed, quality and flexibility.

Since the beginning of the industrial revolution, wide adoption of automation has been hindered by its inherent inflexibility. Historically, capital has been designed to do
something specific and loses a lot of value when process and products change. McAfee and Brynjolfsson [35] have argued thoroughly and convincingly in *The Second Machine Age* that the period of inflexible capital is coming to an end, along with labor's involvement in manufacturing. The author echoes this assessment. MARs are flexible capital, able to be redeployed and redesigned with the fluency of computer code.

The advent of flexible capital marks a dramatic shift in the social constructs that have grown up around manufacturing. Predicting the wider implications for society are well beyond the author's powers, he only offers this note: we are not there yet. Our line employees matter. They matter a lot.

Zeynep Ton [36] makes a strong case in *The Good Jobs Strategy* for many of the suggestions put forward in the latter part of this paper. Her argument, in essence, is that companies that prioritize their employees and have disciplined operations obtain a synergy that equates to a powerful strategic advantage. At its heart, her book critiques the way companies currently prioritize stakeholders. We conclude this thesis by quoting her:

"The good jobs strategy is a long-term investment in your employees with the expectation that those well-paid, well-trained, well-motivated employees will generate even more than they cost. What makes them worth more than they cost is operational excellence."
9. Works Cited


