A Multi-Echelon Supply Chain Model for Strategic Inventory Assessment through the Deployment of Kanbans

by

Philip J. Hodge
Bachelor of Science, Mechanical and Aerospace Engineering, Cornell University, 2001
Master of Science, Manufacturing Engineering, Boston University, 2002
and
Joshua D. Lemaitre
Bachelor of Science, Economics, Duke University, 2001

SUBMITTED TO THE ENGINEERING SYSTEMS DIVISION
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING IN LOGISTICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2008

Signatures of Authors____
Engineering Systems Division
May 9th, 2008

Certified by____
Stephen C. Graves
Abraham J. Siegel Professor of Management
Sloan School of Management Professor of Engineering Systems
Thesis Adviser

Accepted by____
Yosef Sheffi
Professor, Engineering Systems Division
Professor, Civil and Environmental Engineering
Director, Center for Transportation and Logistics
Director, Engineering Systems Division

THIS PAGE IS INTENTIONALLY LEFT BLANK.

A Multi-Echelon Supply Chain Model for Strategic Inventory Assessment through the Deployment of Kanbans

by
Philip J. Hodge
and
Joshua D. Lemaitre

Submitted to the Engineering Systems Division
in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Logistics
at the
Massachusetts Institute of Technology

Abstract

As global competition in the manufacturing space grows, so do corporations' needs for sophisticated and optimized management systems to enable continuous flows of information and materials across the many tiers within their supply chains. With the complexities introduced by the variability in the demand for finished goods as well as by the variability in lead-time of transportation, procurement, production and administrative activities, corporations have turned to quantitative modeling of their supply chains to address these issues. Based on the data of a heavy machinery manufacturer headquartered in the US, this research introduces a robust model for the deployment of strategic inventory buffers across a multi-echelon manufacturing system. Specifically, this study establishes a replenishment policy for inventory using a multiple bin, or *Kanban*, system for each part number in the assembly of products from our sponsors tractor line. We employ a numerical simulation to evaluate and optimize the various inventory deployment scenarios. Utilizing several thousand runs of the simulation, we derive a generalized treatment for each part number based on an econometric function of the parameters associated with lead-time, order frequency, inventory value and order costing. The pilot for the simulation focuses on the parts data for three earthmoving products across eight echelons, but scales to n products across m echelons. Our results show that this approach predicted the optimal quantities of Kanbans for 95% of parts to a level of accuracy +/- 3 bins.

Thesis Adviser: Stephen C. Graves
Title: Abraham J. Siegel Professor of Management, Sloan School of Management
Professor of Engineering Systems Division

Acknowledgements

We would like to thank first and foremost our adviser, Steve Graves, for his consistently steady hand in guiding us through this project. We would also like to thank our primary point of contact at our sponsor company, Dr. Ghulam H. Zahid, for providing insight into the nature of the tractor manufacturing industry and maintaining our sense of direction over the course of the year. Lastly, we would like to thank the Center for Transportation and Logistics for facilitating the partnership with our sponsor company that allowed this project to come to fruition.

We would like to dedicate this thesis to our parents, siblings, and Phil's wife.

Table of Contents

Table of Figures

Table of Tables

1 Introduction

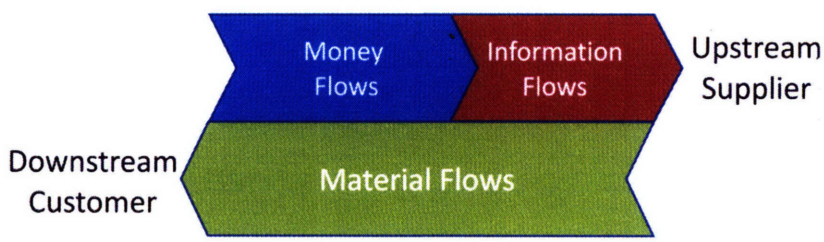
This section has been organized to provide an overview of the concept of supply chain systems as well as the definitions of some of the common nomenclature that will be used throughout this study. First, we will address the status of our sponsor company's supply chain while identifying the various inventory policies it employs or has considered employing. An overview of the thesis objectives will follow, along with a brief review of the published literature on the subject of inventory management in multi-echelon supply chain systems as well as Kanban systems.

1.1 Supply Chain Systems

Any company involved in the production of finished goods operates within what is known as a supply chain system. The concept of a supply chain system (SCS) refers to the techniques used in the planning and execution of procurement, assembly, transportation, and storage of inventory. In the instance of manufacturing, the concept of SCS extends from the acquisition of raw materials to the point of sale of finished products to the end customer, encompassing numerous intermediary production levels along the way. Production levels (also known as echelons, tiers, or stages) refer to any point within the SCS where value is added. So, an echelon can be a workstation, a plant, a machine, or an external company. For this research, we will use "tier", "stage", "echelon" and "production level" interchangeably.

A typical supply chain system involves three primary flows across production levels within the supply chain. Downstream flows refer to those that move toward the end-customer whereas upstream flows refer to those that move toward raw material sourcing. The first flow in a SCS is the information passed upstream between adjacent production levels, originating in the demand signal from end-customers. An example of information flow would be, when an order is placed by an end-customer, an invoice detailing the requested quantity and specifications. The second flow refers to the physical flow downstream of materials in any of the various inventory states, from raw materials to work-in-process (or "pipeline") inventory to final product. An example of this type of flow could be the transportation of raw materials from the point of extraction to the processing plant. The third flow is the exchange of money. For example, as materials are procured from a supplier to an assembly plant, the assembly plant remits payment in exchange for the supplied materials, representing the upstream flow of money. These three flows are illustrated in Figure 1.

Figure 1: Three flows of a supply chain system



1.2 Sponsor Company Overview

Our sponsor company is one of the world's largest manufacturers of construction and mining equipment, diesel and natural gas engines, and industrial gas turbines. The products for the construction and mining industries alone comprise several hundred different models from dozens of product lines with thousands of customizable options, each dependent on the nature of the customer's job. End products for mining and construction range in value from several thousand to several million US dollars per unit.

The company's logistics arm operates out of facilities in the Midwestern United States, and oversees the global operations of over 100 facilities across 6 continents, focusing on production, transportation and distribution planning services. Over the past five years, our sponsor company has achieved overall annual inventory turns, calculated by annual sales over inventory on hand, of approximately 5-7 and operates at approximately 25% gross margins and 10% operating margins.

The domestic operations division, our primary point of contact, is responsible for the production planning for large earthmoving machinery with primary assembly plants clustered in the Midwestern United States. The domestic operations division operates primarily under a Material Requirements Planning (MRP) model, generally known as "push" system. The MRP model is based on a zero buffer concept, whereby all production is made to meet a forecast. Our sponsor is interested in exploring alternative methods of inventory management to determine the strategy that is most sufficiently responsive to the variations in customer demand, production schedule changes, as well as supplier lead-time variability. Specifically, the domestic operations division seeks a more robust inventory management system that will more effectively optimize their replenishment planning and deployment of safety-stock (or "buffer inventory") across the various echelons within their supply chain.

1.3 Supply Chain Overview

This research has a specific application to our sponsor company's SCS associated with the manufacture of large construction and mining equipment in both wheeled and tracked designs. The demand for these items is generated from customers. This demand is then relayed from customers to our sponsor's production planning group through end-product dealers. Our sponsor then sends information for required materials upstream to suppliers, both internal and external. This chain reaction is the information flow shown in Figure 1.

1.3.1 Push/Material Requirements Planning

As mentioned above, Material Requirements Planning (MRP) has been the primary method of our sponsor's production support in recent history, whereby the production planning division develops a forecast for the MRP system to determine part quantities necessary to support the production to meet the demand. This is commonly referred to as a "push" system, since the upstream assembly components are physically pushed through the supply chain according to the forecast. A forecast-driven system such as MRP introduces a phenomenon known as the "bullwhip" effect, where sudden variations in the demand for the end-product exacerbate the variability in production for suppliers, leading to inventory shortages. The resulting shortfalls in production upstream bring the assembly to a halt, resulting in depressed levels of cycle service and increased down-time on the line. To combat this, each part number in the assembly process carries a certain amount of "safety-stock", or buffer inventory, to account for variability in the customer demand patterns and line-side delivery time. This frequently leads to high carrying charges associated with on-hand inventory. This also leads to increased planned production lead-times.

1.3.2 Pull/Just-In-Time Environment

Conversely, a "pull" system refers to demand driven production. When parts are consumed for assembly at Echelon 0, consumption information is passed to the supplier in Echelon 1 to trigger a replenishment. Suppliers then replenish the consumed stock. This system shifts the inventory risk toward the suppliers, reducing the overall system's inventory value and reducing the lead-time variability to that of transportation. Again, this is theoretically based on the premise of zero buffer; however, in practice, safety stock is again typically held in order to combat demand and lead-time variability.

Our sponsors are in the midst of managing a shift away from the purely forecast driven MRP system with the hopes of achieving a pure "pull" system, where inventory is replenished as it is consumed by downstream demand. Currently, their operations are estimated to be at approximately 65% pull with an eventual goal of 80% on a 3-year time horizon (i.e., end of FY2010)[1]. This consumption-based inventory policy will ultimately facilitate a continuous flow of manufacturing that will theoretically improve service levels and at the same time minimize line shut-downs due to inventory shortfalls.

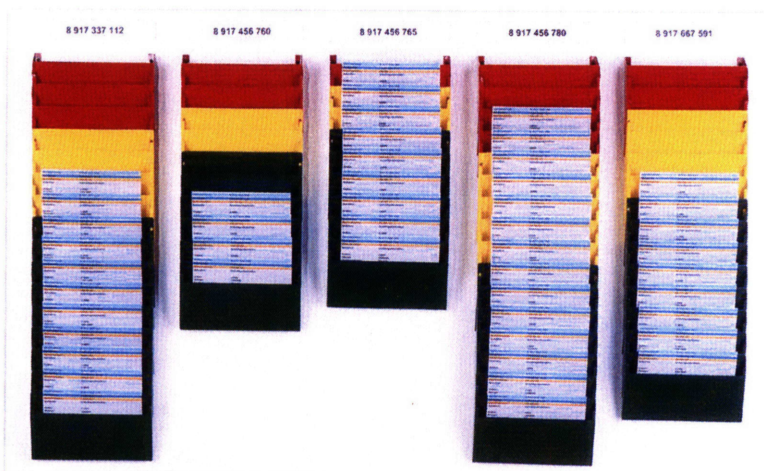
1.3.3 Kanban Overview

One of the primary characteristics of the desired pull system is a bin replenishment system, also known as *Kanban*, from the Japanese term for "sign" or "card". Kanban is a signaling system originally developed by Toyota in the 1950s that triggers upstream production of a part or component once it is consumed in the assembly

[1] These percentage figures are with respect to the dollar value of inventory and not with respect to the stock-keeping unit count.

line. A key enabler of kanban systems is rapid replenishment from the supplier often accomplished by the supplier carrying a level of finished goods inventory in stock ready for shipment. At each stocking point, the inventory level is set based on the expected demand and replenishment patterns. Contributing to the replenishment pattern at the supplier is the utilization level of the supplier production facilities. If the supplier is highly utilized they will have to carry a higher level of finished goods inventory to cover a longer expected lead-time until replenishment. One of the primary theoretical benefits of Kanban is that the lead-time variability is decoupled from the variability of production from the variability in transportation and handling times. Generally the transportation and handling time will be a more stable and predictable quantity to plan around, and the highly variable production lead-times will not affect availability for the end customer.

Figure 2: A Kanban board provides managers with immediate visibility into the inventory status of multiple SKUs using color coded cards. (Source: http://www.shop.org-sys.de)



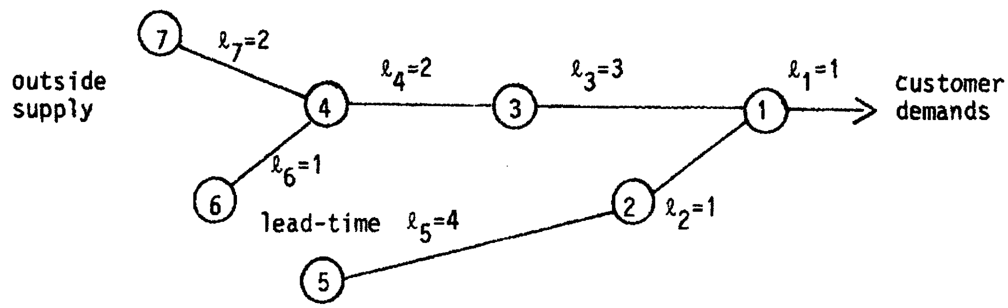
Kanban systems typically use a physical card system, as shown in Figure 2, to indicate the inventory status of individual components (though many companies employ the use of items such as golf balls, or the bins themselves, to signal calls for replenishment). Each card, also known as a Kanban, represents a bin, or a predetermined quantity of part inventory. As a bin is consumed, the empty bin is sent back upstream in the supply chain to be replenished by the part manufacturer. When the bin is returned fully stocked with inventory, the card is replaced to the centralized production Kanban board as shown in Figure 2. When batched and centralized, the Kanban system provides line managers with immediate visibility into the potential pain points of an assembly operation.

Approximately 10% of the dollar value of our sponsor company's inventory is already on Kanban. The success experienced with this inventory is the primary motivation behind management's desires to experiment with full deployment of Kanban across the production of a major line of earthmoving tractors.

1.3.4 Echelon Recognition and Identification

Our research has found there to be no one universal way to identify or label installations within a supply chain system. For example, Clark and Scarf (1960) identify echelon N+1 as one tier upstream of echelon N, whereas Rosling (1988) identifies the individual subassemblies using nodes and inbound/outbound arcs, rather than echelon numbers.

Figure 3: A typical assembly system. Source Rosling (1989).



In general, the literature indicates that any numbering system in a multi-echelon environment should increment as the echelons become further removed from the end product. As such, we used the following numbering convention for the remainder of this study: Echelon 0 is the final assembly plant, with echelon numbers progressing up by 1 with each level removed upstream of echelon zero (i.e., a component for final assembly would reside in Echelon 1; end-product dealers reside in Echelon (1), or negative 1). The suppliers of the parts and subassemblies for the scope of this project operate as many as eight echelons removed from the final assembly, or Echelon 8.

There are various approaches to the recognition of echelons in manufacturing. For example, many supply chains take what is known as a "four-wall" approach to echelon recognition (namely that an echelon only exists if input inventory enters a building and finished goods leave the building). Those with multiple decentralized or independent suppliers use individual companies as the method of echelon recognition, similar to the four-wall approach, only partitioning echelons by building ownership rather than location. So, operations conducted by the same company in multiple buildings would still comprise a single echelon. Graves and Willems (2000) use the following definition to recognize stages:

> A stage represents a major processing function in the supply chain. A stage might represent the procurement of a raw material, or the production of a component, or the manufacture of a subassembly, or the assembly and test of a finished good, or the transportation of a finished product from a central distribution center to a regional warehouse. Each stage is a potential location for holding a safety-stock inventory of the item processed at the stage. (Graves & Willems, 2000)

In the case of our sponsor, for the great majority of the steps in the assembly process, multiple steps are conducted within the same four walls, generating numerous intermediate SKUs along the way. For this research, we reduce the scope of the above definition by restricting echelon recognition to the production tree implied by our sponsor's assembly bill of materials, keeping building codes and parent companies independent of the model.

There are also numerous instances in our sponsor's supply chain where batching or kitting takes place in between the production of components and subassemblies to facilitate higher levels of efficiency on the line. For example, a subassembly requiring ten different SKUs as inputs in various quantities might have these parts picked and organized into a kit prior to arrival on the assembly line. We are not going to be considering these steps in the sequence unless a new SKU/part number is the eventual outcome. And so, the production time associated with kitting will therefore be included in the production time of the next sequential component or subassembly.

1.4 Research Objective

This research has two primary objectives: (1) to develop a general and scalable methodology that determines the safety stock levels in a multi-echelon supply chain through the full deployment of Kanban across all parts and components, and (2) to create a tool that accurately models our sponsor company's inventory management policies by simulating the calculation and coordination of inventory buffers and the respective lot sizes associated with each individual part number. The research element of this project provides a guideline for best practices in our sponsor's organization from the perspective of cost-efficient inventory management through numerically optimized Kanban quantities and sizes. Our sponsor company has specifically tasked us with the development of a prototype for a simulation model that optimizes inventory positioning at the part number level in a Pull/Just-In-Time (JIT) environment[2]. Total cost minimization is the objective function dependent on the quantity of Kanbans and the number of units per Kanban for each part number in the production process. The simulation is constructed quantitatively, based on proprietary company data, and qualitatively, based on employee interviews as well as on the body of published literature surrounding this topic.

1.5 Literature Review

Since Clark and Scarf's seminal work in 1960 on the decomposition of multi-echelon supply chain systems, extensive academic research has been conducted on the subject of manufacturing systems and the inventory policies associated with them. We, therefore, focused our literary survey on published work in two primary areas: (1)

[2] The Just-In-Time principle of procurement indicates that a unit is received from the immediate upstream supplier at the moment it is required in the assembly line.

quantitative modeling of buffer inventory in multi-echelon supply chain networks and (2) the optimization of Kanban/JIT systems. In general, most literature surveyed displayed large amounts of similarity in approaches to dealing with multi-echelon systems, where the primary differentiator from one study to the next was the underlying assumptions that were employed in the model and how they were treated. As such, the value derived from our literature review manifested in extensive consideration for the assumptions and constraints that we would ultimately employ in our own model.

1.5.1 Multi-Echelon Safety Stocks

The crux of this research is the multi-echelon nature of the assembly process in manufacturing systems. We, therefore, began our survey by investigating the extensive body of work on the subject of multi-echelon manufacturing and assembly. A recurring theme in our research has been that the fundamental key to multi-echelon supply chain analysis is the method of generalizing solutions for one echelon to more than one echelon. Clark and Scarf's 1960 paper is the previously cited seminal work on this subject and lays out the framework for how induction can be used to generalize from one echelon to two echelons and, therefore, to an arbitrary number of echelons. Our modeling efforts were guided precisely by this decomposition approach, beginning with a single part number and gradually incrementing the level of complexity to capture assumptions and business rules effectively so as to accurately approximate our sponsor's assembly sequence. As the first major paper on multi-echelon systems, Clark and Scarf (1960) also define the framework on which most subsequent papers are based.

Multi-echelon supply chains can be classified into two categories: (1) inbound chains supplying a production facility and (2) outbound distribution chains delivering products to customers. Although our research is focused on inventory allocation across the inbound supply chain, there are insights and methodologies, such as techniques for modeling the network and simulating material flow found in Bookbinder and Heath (1998), which have lent significant value to our simulation prototype. Bookbinder and Heath's paper specifically deals with the lot-sizing question in a distribution requirements planning environment, where the rolling schedule is the primary differentiator from prior work. As one of the primary goals of our prototype is to deliver a solution that will recursively answer the uncertain demand question over the long-term, this environmental condition of the rolling schedule is critical to our research.

The focal point of our research is the allocation of buffer inventory. Specifically, we aim to identify and optimally treat the points in the supply chain network that run the risk of shutting down the entire process in the event of a stock-out. Graves and Willems (2000) employ a digital camera assembly as a case study to illustrate the situation of a guaranteed level of service to customers under bounded demand. They further introduce the concept of decoupling points in the process. This plays a significant role in our research, especially in reference to the part numbers common to numerous assemblies, intuitively the most likely to effect line delays. Furthermore, the bounded demand assumption is implied by the schedule freeze practiced by our sponsor.

The paper entitled *Safety Stocks in Manufacturing Systems* (Graves, 1988) provides one of the more comprehensive literature reviews on the subject of inventory planning in multi-stage manufacturing systems and further develops a model flexible enough to handle centralized and decentralized inventory controls. Lee and Billington (1993) build on Graves' model by applying it in the context of a decentralized assembly system to evaluate various alternative supply chain designs for a Hewlett Packard Deskjet model printer. The domestic operations division of our sponsor follows a centralized control system; however, the scalability of our model to decentralized systems addresses the research goal of this study to extend beyond the focused application to sponsor data. As such, these papers contributed to the overall flexibility of our model.

As discussed above, the negative effect of safety stock is largely the costs associated with holding additional inventory. As such, Diks and de Kok's 1999 paper was highly relevant to our handling of inventory costs. They begin with the decomposition of a generalized arborescent N-echelon network, similar to that of Clark and Scarf (1960), and algorithmically optimize using a three-echelon simulation under the assumption of level demand with a uniform distribution. While Diks and de Kok's model is based on a divergent distribution system as opposed to a convergent assembly process like that of our sponsors, they raise the substantive question of inventory allocation between stations within a single echelon. Reverse engineering the allocation problem to an assembly network becomes relevant to our research due to the significant proportion of parts common to multiple subassemblies and ultimately end-product models. Mittra and Chatterjee (2004) similarly optimize their system via the development of mathematical models.

Simulation modeling is the primary experimentation method in the research space of multi-echelon supply chain. Works such as Lagodimos and Anderson (1993), Tee and Rossetti (2002) and Axsäter (2000) all provide simulations as the primary driver of results. Stenger (1996) employs a regression-based simulation to evaluate inventory issues for a ceramic flatware manufacturer through iterative experimentation under variable conditions.

Rosling (1989) also takes an analytical approach to the demand variability problem, only in assembly systems. Also based largely on the work of Clark and Scarf, Rosling's paper decomposes the assembly system into a serial system such that it can be generalized to an arbitrary number of echelons. Rosling generates his model contingent on the sequence of events beginning and ending at the start of the period. This is a simplification of the system that we have adopted in modeling our sponsor's assembly process. One distinction from Rosling's assumptions is that we have chosen not to include inventory already assembled into other units; this outlines the difference between forecast based studies and pull/JIT environments. The work in progress inventory is not considered in our model since bins are replenished from upstream finished goods inventory as they are consumed, independent of the production time that typically drags down performance in push systems.

Bollapragada et al. (2004) is one of the most commonly referenced studies of multi-echelon modeling in the space of assembly networks. They take a two-echelon approach to treating uncertainty in end-product demand and lead-time variability, employing computational experiments to determine supply chain improvements with respect to safety-stock costs, similar in many respects to our approach. Their findings validate our sponsor's shift toward the characteristically reduced lead-times in pull/JIT systems, as the cost benefit is determined to be correlated to the relative value of the part or component; our project, while inclusive of the majority of the input value spectrum, is indeed focused on the later stages of the supply chain where the inventory value per unit of input is higher.

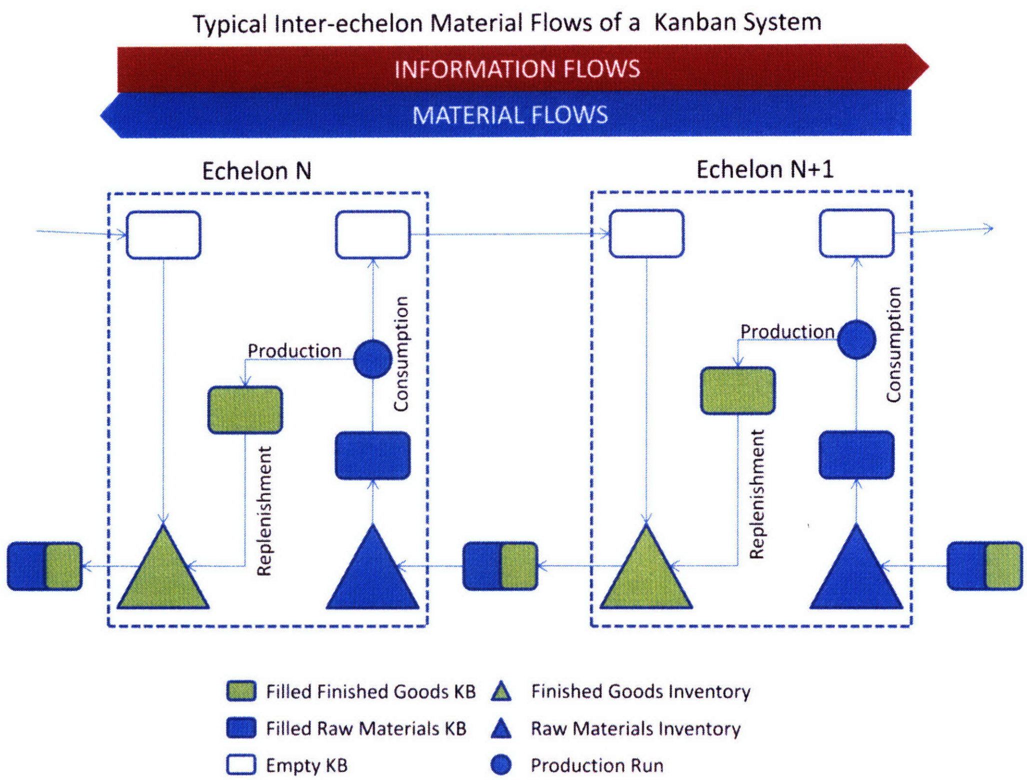
1.5.2 Kanban/JIT Manufacturing Systems

The second body of literature that we investigated was on the subject of the deployment of Kanban/pull systems. Since the founding of the concept by Toyota in the 1950s, manufacturers around the globe have had a vested interest in evaluating the potential application to their specific product lines. As such, there is a vast body of work on Kanban, ranging from qualitative treatments assessing the appropriateness of integration in existing business models to specific case studies dealing with the quantitative simulation and optimization, much like the problem with which we are faced. As an introductory reference point, Esparrago (1988) provides the most concise overview of the concept of Kanban, including the historical origins, benefits and varieties of implementations across numerous cited Japanese manufacturers. Zaenglein (2000) provides a more thorough history of the evolution of Kanban as well as a detailed description of the execution of a Kanban system, with specific application to the automotive industry. We recommend these papers as a primer to readers unfamiliar with the subject of Kanban.

Since our research hinges on the benefits of Kanban with respect to costs, we restricted the scope of our literature survey to prior work conducted in the space of mathematical simulation of Kanban systems. Deleersnyder et al. (1989) takes the study of Kanban to an extensive degree of detail with specific emphasis on the "operational control" problem, or determining the appropriate quantities of bins and where bins are to be allocated throughout the assembly process in an effort to combat the inherent issues of variable demand, one of the primary production disturbances addressed by our research. This work represents one of the first analytical and quantitative approaches to Kanban system modeling. Subsequent quantitative work includes Nori and Sarker (1998), Köchel and Nieländer (2002), and Gurgur and Altiok (2004).

Optimization studies lend further credence to the value of Kanban systems as a viable alternative to MRP. Wang and Sarker (2005) take a mixed-integer nonlinear programming (MINLP) approach to modeling the Kanban system, using the total system cost as the objective function and introduce the queuing concept to address the problem of container quantities. Xiabo, Gong and Wang (2002) also follow Deleersnyder's study of operational control while explicitly separating input buffer from output buffer at each stage in the assembly process, whereby an emptied input Kanban in a stage's input buffer signals a pull from the adjacent upstream stage's output buffer, as shown in Figure 4.

Figure 4: This figure illustrates the typical flow of material in a Kanban system. Note that finished goods inventory in Echelon N+1 is the same as the raw materials inventory in downstream Echelon N.



One of the differentiators of our research from most other work on the subject is our use of regression analysis following experimental runs of a simulation to generate reasonable and scalable parametric treatments of parts in our sponsor company's supply chain. Jothishankar and Wang (1992) take a different approach to the simulation of a Kanban system by utilizing linear regression metamodeling to describe the relationships between both quantitative and qualitative factors on the overall system. This provided a precedent for the type of variables that we would consider in our own case study as well as an experimental design to generate adequate predictors while reducing the number of simulation runs required. Due to the large number of part numbers in our assemblies, this approach presented itself as a viable option, especially when considering the range of parameters and the eventual application to larger quantities of end product models.

As mentioned previously, our project models the production of three end-products. Singh et al. (1989) uses a similar approach of three items. They use a fixed bin size equal to ten percent of the expected daily demand and a fixed bin replenishment time, limitations based on prior research conducted in reference to Toyota. They further restrict the model flexibility by fixing the daily production. Takahashi and Nakamura (2002) present a much more reactive and decentralized model wherein the buffer sizes can adjust dependent on the systematic instabilities, validated by a series of simulation experiments. These instabilities are accounted for by randomly distributed lead-times and demands, the seminal assumptions of our research.

1.5.3 Literature Summary

This literature review captures the primary published work on the subjects of multi-echelon supply chains and Kanban implementations that we found to be relevant to our research. Additional surveyed literature can be found in this paper's bibliography. In the end, our final simulation model was not derived directly from any previous research as the requirements of the project were very specific to the nature of our sponsor's operations. However, our review provided several critical insights in the formulation of our assumptions as well as our treatments and inclusions of business rules. Our use of simulation-based experiments as inputs to a linear regression to generalize the treatment of part numbers within an assembly system marks an alternative approach that we believe ultimately generates effective and easily replicable recommendations for both ad hoc and system wide optimization exercises in Kanban implementations.

* * *

In the remainder of this paper, we provide a detailed introduction to project specifics, including assumptions, constraints and other environmental issues specific to this research. We then provide a detailed description of the iterative construction of our simulation model. Finally, we conclude this paper with an analysis of the results followed by specific recommendations for our sponsor's specific inventory policies as well as for future research on this subject.

THIS PAGE IS INTENTIONALLY LEFT BLANK.

2 Understanding the Problem

Before delving into the details of the model, it is important to gain a deeper understanding of the motivations behind our research. As such, we use this section of the paper to discuss the research question with respect to the environmental characteristics of our sponsor company. The information presented in this section is based on extensive interviews with company personnel and on-site facility visits. We then discuss the data provided by our sponsor and the manner in which we specifically use it. This data includes the Bill of Materials (BOM), the relevant plant and product line production history, cross-model parts (parts required by multiple product models), as well as a decomposition of lead-time as defined by our sponsor. We further identify the business rules and assumptions that require consideration in the development of the model. Lastly, we conclude with a qualitative discussion of the various components of the costs included in the objective function.

2.1 Problem Definition

As mentioned earlier, this research is intended to determine a best-practice guideline for the deployment of Kanbans as evaluated by extensive experimentation through an exercise in simulation on actual company data that is to be used by managers of manufacturing systems similar to those of our sponsor company. The primary output for this research is a functioning model prototype that effectively mimics the assembly process along with the variability that is introduced by both lead-time and demand.

2.1.1 Focus

We build the model around the parts data for three earthmoving products produced in three primary Midwestern manufacturing locations, to be referred to as Model 1, Model 2 and Model 3 for the duration of this paper. As mentioned earlier, our research extends to suppliers that are as many as eight tiers removed from the final manufacturing operation. The project takes into account variability in customer demand, lead-times, quality, service level and production volume. We are not evaluating the locations or number of facilities, as such questions of network design are beyond the scope of the project; nor are we evaluating changes to parts and sourcing of raw materials as they too are topics beyond the project scope.

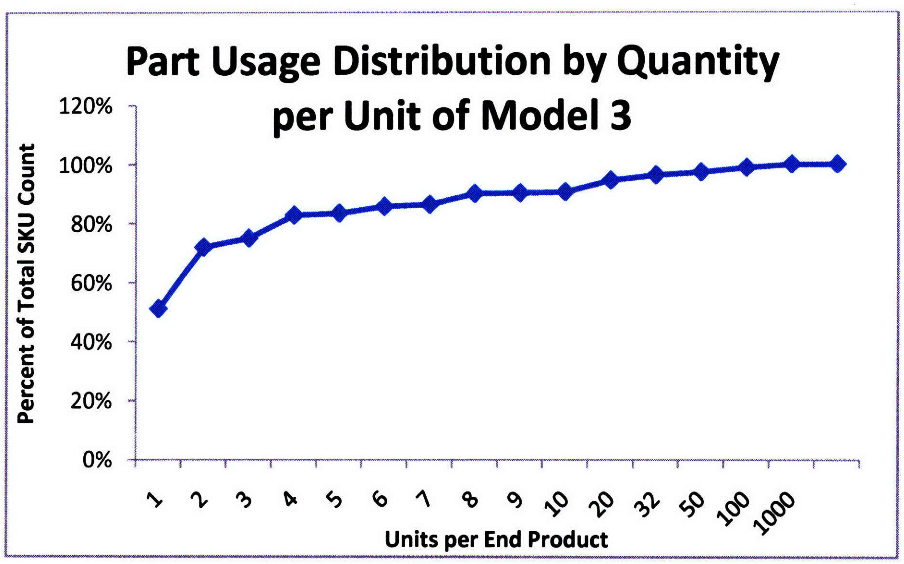
2.1.2 Data Description

The data driving our modeling efforts is restricted to the production of three models, Models 1, 2, and 3, manufactured in the assembly facilities near the domestic operations division headquarters. The primary focus is on the Model 3, the most valuable and high-margin of the three; however, the parts required to build Model 3 significantly overlap with those required to build the other two. As such, these models are also included to account for the cross-demand variability of part numbers with multiple end-products.

2.1.2.1 Bill of Materials

We are utilizing the Bill-of-Materials (BOM) for Model 3 to map the entire build sequence from raw materials to final production. A BOM is a list of materials, parts and components required for the assembly of a manufactured item. In the case of this particular line of tractors, uncustomized assemblies require approximately 1500 part numbers in quantities varying from one to over 900, introduced in as many as 4 echelons. The BOM specifically identifies the part number, echelon number, quantity required and the subsequent part number, hence providing us with a means for symbolically developing the part dependence tree for the relevant products. As such, this is the source of the large majority of the substance for the model. Figure 5 shows the cumulative distribution function of the various part numbers required in a Model 3 tractor, with over 50% of all part numbers requiring a single unit per end product and 82.8% requiring 4 or fewer units.

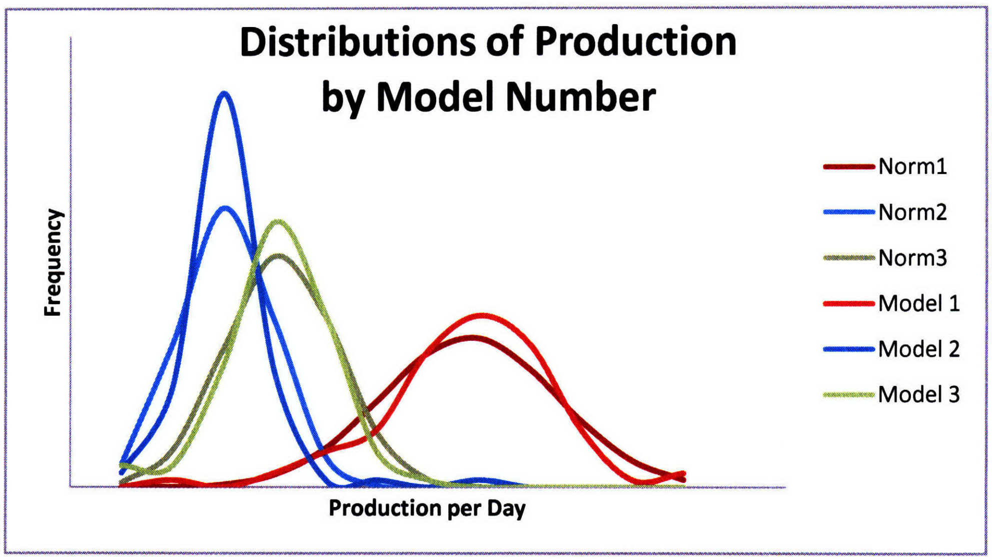
Figure 5: The cumulative distribution function of components of Model 3.



2.1.2.2 Historical Production and Demand Planning

Our sponsor company has also provided us with detailed production history data for the relevant line of tractors manufactured in the Midwestern facilities. The data includes the quantities and models of tractors completed and approved by calendar date for the trailing three month period. We are using this data to approximate the mean expected demand and standard deviation for each of the three relevant models over time. This is used to estimate the distribution of daily demand of each end product. By plotting the production history for the various models, we estimated a normal distribution for each of the models, as shown below in Figure 6.

Figure 6: The distribution of daily production by model plotted against the respective normal distribution functions.



We note that this is merely a means to achieving an approximation of demand variability. While the production is not necessarily indicative of the actual experienced demand, it should be adequate to provide us with a reasonable order of magnitude for the actual demand variability. We are assuming that there is no partial production on a given day (i.e., there cannot be a fraction of a tractor built at the end of the production day). Any missed production will be added to the build schedule for the following production day.

2.1.2.3 Inter-Model Parts

We have also been provided with the part numbers common to the multiple end-product models relevant to this research. As mentioned above, many part numbers involved in the assembly of the Model 3 are also used in the assembly of other end-product models. We are using this information to more accurately replicate the amount of buffer inventory necessary to satisfy the demand variability of all three models.

2.1.2.4 Order Costing

As the objective of this research is to determine the Kanban quantities that optimize the total cost across the entire assembly system, a critical component of our success measurement is the cost associated with ordering product between echelons. We are using our sponsor company's best approximations for the order costs for the procurement of each part number at each echelon.

2.1.2.5 Decomposition of Lead-Times

We have also been provided with the lead-times for all parts required for Model 3. This has been broken down for us as follows:

- *Order issue lead-time* – the time from the point when the order is placed to point when the order is processed.
- *Supplier time* – time required for a supplier to prepare the part for transportation loading (i.e., picking, intra-facility moves); this does not include any stage in the manufacturing process.
- *Transportation time* – dock-to-dock time

- *Distribution time* – time from cross-dock to point of use, such as a storage or staging area.
- *Indirect Process Time* – any additional time required by our sponsor to ready product for consumption.

All times have been provided as integer dates (i.e., there are no timestamps in the data to account for time of day when an item is received). The mean and standard deviation of lead-time for each product are based on the most recent series of 30 receipts for each part number. Data has been manually collected and input by sponsor employees as well as external suppliers; because of this, we can assume that it is prone to error. An analysis of variance revealed a clear need for some rudimentary outlier trimming, primarily on the high end of the spectrum as the low end is generally bound by zero days. Table 1 provides some summary statistics for the various lead-time classes. The numbers have been masked for confidentiality reasons but are indicative of the scale of the actual numbers.

Table 1: Summary Statistics for the various components of sponsor's lead-times

	average	min	max	StdDev
Order Issue	1.06	1	11	0.42
Supplier	8.71	0	95	7.91
Transportation	3.67	0	38	5.93
Indirect	0.02	0	5	0.30
Distribution	2.11	0	9	0.49
Total	13.64	3	108	10.06

As mentioned earlier, we are assuming a Kanban environment where upstream inventory is ready for replenishment as triggered by consumption in downstream echelons. Therefore, the lead-times associated with filling orders should theoretically be contained in these lead-time designations. There is some dispute with regard to the validity of the Supplier Time being exclusive of the actual production associated with respective part numbers, since on the maximum extreme, it would be highly unlikely for a supplier to require 95 days to pick and pack an item. Nevertheless, we are assuming that this is a practical case.

2.1.3 Assumptions, Business Rules and Constraints

In an effort to best understand the issues that a manufacturing organization faces, we have conducted extensive interviews with logistics personnel and plant managers, both on-site and via teleconference. Subsequent to our interviews with our sponsor company's staff and our visit to the assembly facilities, we have developed a working list of assumptions that are specific to our sponsor company.

Improved performance in the JIT environment: JIT material management has led to improved levels of stock availability and decreased levels of line shut-downs in trial implementations. Our sponsor company wants to move in the direction of a pull-based material management system away from the historical MRP/push system. It is not in the scope of this project to evaluate the relative performance of these systems, but rather to conduct an experiment in the environment of Kanbans.

Uniqueness of supplier: Products and subassemblies are unique and come from unique suppliers. This means that upstream inputs in the supply chain can come from one and only one supplier.

Commonality of parts: Parts and components can be included in the assembly of several end products.

Transportation costs: Transportation costs will be assumed to be included in the cost of ordering replenishments. In the case of our sponsor, the transportation between the facilities relevant to the tractor models that we are studying is small enough in distance and cost to be regarded as negligible. However, the lead-times associated with transportation will be included in the model and hence provide an implicit cost associated with transportation.

Carrying cost of inventory: The holding costs are estimated by management to be 11% per annum. This is an average cost of capital and, therefore, will not reflect the estimated return from the next best alternative to an investment in inventory for the part number in question. For the sake of simplicity, we are adjusting this to 12% in the event that a month-by-month analysis is requested by our sponsor.

Units per bin is unconstrained: In practice, some of the later echelon components, such as 20,000 lb frames and engines, are unlikely to be shipped in increments other than a single trailer load. Conversely, C-items such as nuts and bolts are likely to be shipped in lots on the order of thousands simply due to the relatively low inventory value of these parts. We do not have any upper/lower bound data on the typical shipment size of parts. In the absence of these sku-by-sku constraints, we are assuming that there is no constraint to the quantity of units per Kanban.

Production shortfall rolls over: Missed production gets rolled over to the following production period.

Additional shifts: As the production shortfall accumulates, an additional shift will be tacked on to the end of the production week, but only if a predetermined threshold of backorders has been breached. This is captured in our model by a Saturday run. The threshold can be set and altered by management. For example, with a hurdle of 5 units, if the number of backordered units is at least 5 on Friday, the time available for production would be increased by the length of a weekend shift (by default equal to one 430 minute weekday shift). The amount of production would still be calculate based on the parts and production time available the same as a regular shift.

Machine reliability: Machine reliability is assumed to be 100%. This means that no variability is introduced by the availability of production facilities (Deleersnyder, Hodgson, Muller, and O'Grady, 1989).

Shift length: We assume for a single eight-hour shift per day, or 480 minutes; when factoring in breaks and lunch, total production time per day is equal to 430 minutes. A production week equals five production days of 430 minutes. Additional shifts (Saturday runs) are introduced to the model if the backordered quantity exceeds a specific threshold. This simply extends the Friday shift to 860 minutes.

Production time is non-constraining: The manufacturing process is not constrained by production time; i.e., production can only be constrained by parts availability. To accommodate the assumption of non-constraining production time, we increase the production shift length if a particular day's mix would require more production than a standard shift length. We do not decrease the production shift length if the mix is shorter than the extra time available for producing backordered tractors; in this case, the remaining production time would simply go unused. This study is strictly an experiment of the impact of inventory policy on production performance.

Production lines: Only one production line is used. This should not limit the scalability of our model to production systems with multiple lines running in parallel.

Transfer of ownership: Ownership of a part does not occur until the part arrives in inventory. This consideration is directly relevant to the calculation of inventory carrying costs.

Distribution of Demand: Based on the historic distribution of production, we are assuming that the daily demand is normally distributed about the production means and standard deviations as derived from the trailing 90-day production schedule provided by our sponsor. See Figure 6.

Lead-Time Distribution: Lead-times are also assumed to be normally distributed based on means and standard deviations provided by historic receipt data.

2.2 Decomposition of Costs

The ultimate motivation behind our research is our sponsor's interest in cost reduction. Amidst the burgeoning environment of global competition in manufacturing, corporations around the world have had a vested interest in reducing the cost of operations in order to advance, or even maintain, their market position. One of the major components of these costs is accrued in corporations' inventory accounts. These accounts are comprised of cycle stock, in-transit (or "pipeline") stock, and safety stock. Cycle stock refers to the component of ordered inventory that is a fixture in the replenishment of the item. So, when a replenishment order is received for an item, the cycle stock is intended to cover the expected demand over the order cycle, namely the expected demand until the next order arrives. The safety stock, as mentioned earlier, refers to the inventory component accounting for any demand variability that arises between when an order is placed and when the order arrives, such as demand spikes, transportation lags, or line shut-downs. The in-transit stock, as the name implies, is simply the inventory that has already been ordered and is en route to the next downstream echelon.

2.2.1 Order Costs

When a bin for an item is consumed in a Kanban environment, an order is triggered for the replenishment of that bin. Because of the administrative costs associated with processing, preparing, packing and shipping the order, an order cost is tied to the individual invoice. This cost is fixed with respect to the number of Kanbans replenished in the single order, as we are assuming that the capacity for in-transit inventory is unconstrained. For example, the order costs associated with a single bin replenishment are the same as those associated with a ten bin replenishment.

2.2.2 Procurement Costs

Not to be confused with order cost, procurement cost is the cost of the inventory ordered. To reuse the example from §2.2.1, a replenishment order of ten bins will have a procurement cost ten times that of a single bin replenishment order, assuming no volume discount.

2.2.3 Holding Costs

Typically, companies will designate a holding charge for inventory since it represents money that could have been allocated elsewhere in the business or in external investment markets. For example, a company may have an average of $100 million tied up in inventory over the course of a year accruing holding charges of 15% per year, or $15 million per year.

2.2.4 Shortage Costs

A backorder is the event when the available stock fails to satisfy the demand. For example, if a tractor frame is ready for an engine assembly to be installed but there are no engines in available inventory, the demand for engine installations cannot be satisfied, hence yielding a backorder. In the event of a backorder, shortage costs are incurred. Typically, backorders are added to the following production period's schedule. Because demand is not met, a downstream sale is lost or delayed and the production line for that item might stop or shut down, both incurring potentially significant opportunity costs. Provided these can be accurately monetized, a shortage cost is incurred to represent these opportunity costs. Our sponsor has minimal data in this regard. As such, we are selecting rule-of-thumb estimations for shortage costs in determining the cost of the overall system's production.

3 Methods: Numerical Simulation Modeling

In this section, we discuss the development process of the simulation model that provided the backbone for our analysis. We provide an overview of the underlying structure of the model. We then discuss the various components, individual inputs and intermediate calculations. Finally, we introduce the total relevant cost function on which the model outputs are measured. All models are designed to simulate a single part at a time.

3.1 Model Structure

The key tool for our investigation is a model to complete time-series simulation of the production system. Our model simulates the daily manufacturing production, tracks part consumption and then places and tracks future orders inbound to the system. In this section of the paper, we first explain the overall structure of the model and then the key performance statistics used to compare the results. Finally, we include a detailed discussion of the specific variables and calculations used to implement our assumptions and production rules into the model. The simulation is used to evaluate the performance of the system under different parameters. Following from our focus on inventory placement in the system, our model is particularly focused on tracking which parts are available for production and on order. To compare the results from various inputs, we use the total relevant cost consisting of carrying cost, order cost and backorder cost. Our model was prototyped in Microsoft Excel and then transferred to MATLAB to perform the majority of calculations.

There are two aspects to the structure of the model: the conceptual theory around which the simulation is built and the bookkeeping scheme used to track the data. This section is primarily focused on the conceptual theory while the bookkeeping is more substantially handled in a later subsection of the methods. In developing the model, we found four key areas that needed to be handled: (1) what needs to be built, (2) what can be built, (3) what is built or consumed and (4) what is on order.

The requirements for what *needs to be built* each day are calculated based on the number of units backordered from the previous day plus the daily production schedule. The number of units backordered is the total number of units required the previous day that were not built due to inventory shortages. This figure is not precisely the same as the number of *backorders* we track for total cost because if a unit is backordered, it might be carried on the rolling requirements list for multiple days until it can be completed, either during a regular shift or by scheduling an extra Saturday shift. To capture the effects of variable production schedules, we model the production schedule of each tractor model as a normally distributed variable with a nominal average and standard deviation based on historical data. For the simplicity of our calculations, the production schedule of each tractor model is distributed independent of the other models.

Based on the production requirements, we then calculate the number of units that *can be built*. There are two constraints on production simulated in the model: the primary constraint is the number of parts available for production and the secondary constraint is the amount of production time available. The usage in parts-per-unit by tractor model is one of the inputs to the model and can be changed to investigate the dynamics of different parts. In the situation where there are not enough parts to complete the total production requirement, production is prioritized for tractor Model 3 then tractor Model 2 and finally tractor Model 1. This is the same way units are prioritized if there isn't enough time to complete the required production because the relative value and margin on these tractor models increase with the model number.

Based on the parts available for production, the simulation model calculates the "part-limited" production, representing the maximum production level that can be completed up to the total production requirement. This "part-limited" production is used as the basis for calculating the "time-limited" production, representing the maximum production level that can be completed based on the time available up to the "part-limited" level. While the quantity of parts in inventory available for production is simply calculated based on consumption and receipts, the time available for production is a more complex calculation.

To accommodate our simplification that the production schedules are independently distributed, we may produce schedules that are not feasible; therefore, we change the length of each production shift to be the greater of (a) the baseline 430-minute shift and (b) the total time required to build the production schedule based on the time per unit. For example in our standard system, the production schedule for model 1 ranges between 5 and 10 units per day, model 2 ranges between zero and 4 units per day and model 3 ranges between 3 and 5 units per day. When the production schedule is 7, 2 and 3 for model 1, 2 and 3, respectively, the production time is completed within the standard shift. However since the production schedules are distributed independently, a production schedule of 10, 4 and 5 could be selected which would take longer than a full shift to complete. In these situations the shift length is extended to equal the time required. In addition to the daily production shift length, if there are a sufficient number of backordered units, additional production time representing "Saturday" shifts is added to the available production time.

The "time-limited" production represents the maximum production possible that day. The "time-limited" production is directly calculated from the "part-limited" production, the production time per unit and the total production time available. Our model assumes that this "time-limited" production will be produced everyday. Any other reductions based on machine breakdowns or other external factors are handled by the variability introduced into the production schedule based on historical data. The number of parts consumed from inventory is calculated based on the number of parts consumed per unit of production. Because our model is focused on the use of bins to manage inventory in the system, we also calculate the number of bins that are at least partially full at the start and end of the day to track how many were emptied. The number of bins emptied is tracked for replenishment as essentially an empty bin becomes an order for replacement. In a real world system, orders will not always be immediately forwarded on to suppliers, and requirements may be batched together to place fewer larger orders. To account for this characteristic, our model also handles the complexity of limiting the number of orders per 20-day cycle.

Tracking the inbound orders takes up almost half of the bookkeeping in the model but is based on relatively simple calculations. We assume the lead-time of new orders to be normally distributed with some known average and standard deviation; this treatment is driven by the assumptions that (a) the lead-time primarily represents the transportation time and (b) the supplier will keep finished inventory on hand for replenishments. The lead-time is calculated from the normal distribution based on a uniformly distributed random seed. Each day, the orders outstanding are updated by shifting the previous day's orders by one day and adding any new orders placed today (i.e. orders 8 days out on day 1 are 7 days out on day 2). The orders one day out are assumed to have arrived and are dropped out of the inbound tracking system. This method is enabled by calculating the lead-time for each order when it is placed according to the known distribution then keeping it fixed because we have already accounted for the variability.

3.2 Equations, Objective Function and Constraints

Our model is built around a large matrix where each column represents a particular variable and each row represents a different day in the simulation. The flow of calculations through the model follows the sequential completion of the calculations for all of the variables on a given day (i.e., a given row) before moving on to the next day (i.e., the next row). Variability enters the production system through fluctuations in the lead-time and production schedule; our model captures that variability using probability distribution functions and samples from random seeds.

3.2.1 Nomenclature

The matrix we use to track our model grew in complexity and size through multiple iterations. We ended up with a model 88 columns wide with variables grouped by what they represent and when they were used in the calculations. The positions of some of the variables look sub optimal when reviewed; however most of them allowed for more clarity while debugging and developing the model. While each column represents a different variable each row represents a different day of the simulation. We ran our simulation over 5000 days representing 20 years of 250 production-days per year.

$$Model = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix}$$

$$n = 88\ columns$$

$$m = \#\ of\ Days$$

Each column of the model represents a different variable in the state of the system on that particular day.

As explained in the first section of the methods there are four major categories into which the calculations and variables can be grouped. Those categories are:

1. what needs to be build
2. what can be built
3. what is built or consumed
4. what is on order

Here are the columns of the matrix grouped by category. Columns 42 and 85 are kept blank to allow for future expansion of the model without having to expand the matrix.

3.2.1.1 What needs to be built

$$a_{x,1} = Day$$

Day represented by this line of the simulation, for the base models this runs from 1 to 5000. To calculate which schedule seed to use we rely on knowing which day we are simulating.

$$a_{x,2} = \text{LT Random Seed}$$
$$a_{x,3} = \text{Model 1 Random Seed}$$
$$a_{x,4} = \text{Model 2 Random Seed}$$
$$a_{x,5} = \text{Model 3 Random Seed}$$

The seeds from uniform distribution between zero and 1 used as the input for inverse probability distribution functions when calculating random factors of the calculations. To keep the experiments repeatable these are sampled out of a known set of random variables.

$$a_{x,6} = \text{Production Schedule Model 1}$$
$$a_{x,7} = \text{Production Schedule Model 2}$$
$$a_{x,8} = \text{Production Schedule Model 3}$$

The production schedule for each day is calculated based on the random seed and the production distribution characteristics for that model. We assume the production schedule to be normally distributed based on an average of the nominal schedule and a standard deviation based on a histogram of historic production numbers.

$$a_{x,9} = \text{Model 1 Daily Production Requirements}$$
$$a_{x,10} = \text{Model 2 Daily Production Requirements}$$
$$a_{x,11} = \text{Model 3 Daily Produciton Requirements}$$

The daily production requirements for each model are based on the production schedule and any units that have been backordered. The daily production requirement is calculated based on the sum of the previous days production requirement plus the current days production schedule minus the previous days production.

3.2.1.2 What can be built

$$a_{x,12} = \text{Inventory On Hand}$$

Each day the inventory on hand is calculated, and represents the total number of parts available for use in production on a specific day. The inventory on hand is calculated as the previous days inventory on hand minus the inventory consumed during production plus the parts received today.

$$a_{x,13} = \text{Total Part Requirements}$$
$$a_{x,14} = \text{Part Requirements Model 1}$$
$$a_{x,15} = \text{Part Requirements Model 2}$$
$$a_{x,16} = \text{Part Requirements Model 3}$$

Each day the total number of parts required to build all of the units of the daily production requirements is calculated. The requirements for each model of tractor are also calculated. Each of these is the product of the daily production requirements and the number of parts per unit input to the system.

$$a_{x,17} = \text{Production Limited Flag}$$
$$a_{x,18} = \text{Units of Model 1 to Build}$$
$$a_{x,19} = \text{Units of Model 2 to Build}$$
$$a_{x,20} = \text{Units of Model 3 to Build}$$

Each day based on the number of parts available and the daily production requirement the model calculates if the number of parts available would limit production.

If production is limited, the model calculates the number of units of each model to produce, based on a prioritization of Model 3, Model 2 then Model 1 attempting to use up as many parts as possible. If the production is not limited by the number of parts available, the number of units to build is set equal to the daily production requirements.

$$a_{x,21} = \text{Saturday Production}$$

In order to simulate the actual production system, the model takes into account the fact that a shift can be added on a Saturday if there are enough backordered units. The calculation of Saturday production is made if the number of backordered units is over a "hurdle" rate input to the system and the particular day being simulated is a Friday. The backordered count is based on the rolling production requirements in columns 10 to 12 compared with the daily production schedule in columns 6 to 9.

$$a_{x,22} = \text{Production Minutes Available}$$

Each day the number of production minutes available is calculated based on the length of shift and if there is a Saturday shift (for the ease of calculations Saturday shifts are simulated by adding production time onto Friday shifts).

$$a_{x,23} = \text{Total Minutes Required}$$
$$a_{x,24} = \text{Minutes for Model 1}$$
$$a_{x,25} = \text{Minutes for Model 2}$$
$$a_{x,26} = \text{Minutes for Model 3}$$

Similar to the number of parts required for production, the number of production minutes required to complete production is calculated. The parts limited production is used as the basis of this calculation since the number of units built is constrained by the number of parts available. Both the total for all models and the specific quantities for each model are calculated.

$$a_{x,27} = \text{Time Limited Production Model 1}$$
$$a_{x,28} = \text{Time Limited Production Model 2}$$
$$a_{x,29} = \text{Time Limited Produciton Model 3}$$

If there is not enough time to complete the entirety of the part limited production, production is prioritized for Model 3, Model 2 then Model 1. If there is sufficient time to complete all production then the time limited production numbers will be equal to the part limited production quantities.

$$a_{x,86} = \text{Production Loss Model 1}$$
$$a_{x,87} = \text{Production Loss Model 2}$$
$$a_{x,88} = \text{Production Loss Model 3}$$

For each model the production losses are tracked based on the actual production numbers and the production schedule. Production losses are tracked as back ordered units and added to the rolling production requirements.

3.2.1.3 What is built or consumed

$$a_{x,30} = \text{Production of Model 1}$$
$$a_{x,31} = \text{Production of Model 2}$$
$$a_{x,32} = \text{Production of Model 3}$$

These variables represent the actual production numbers for each model. The production level for each model is bounded by the time-limited production for each model, but could be modified if there were other constraints. For our model we took the actual production to be equal to the time limited production.

$$a_{x,33} = \text{Parts Consumed}$$
$$a_{x,34} = \text{Ending Inventory}$$

The number of parts consumed each day is calculated based on the units of production of each model and the number of parts per unit. Ending inventory is calculated based on the beginning inventory and the number of parts consumed by production.

$$a_{x,35} = \text{Stating Number of Bins}$$
$$a_{x,36} = \text{Ending Number of Bins}$$
$$a_{x,37} = \text{Bins Consumed}$$

The inventory control model is based on *Kanban* bins used to manage and control part inventory. At the start and end of each shift we calculate the number of bins at least partially full based on the starting and ending inventory and the number of parts per bin. Based on the starting and ending number of bins we calculate the number of bins consumed or emptied. Empty bins become the orders for replenishment from the suppliers.

$$a_{x,38} = \text{Rolling Bin Requirement}$$
$$a_{x,39} = \text{Days Lapsed Since Order}$$
$$a_{x,40} = \text{Place Order Flag}$$
$$a_{x,41} = \text{Order Quantity}$$

The rolling bin requirement represents the number of bins required for replenishment that have not been ordered yet. The requirement is calculated based on the previous day's requirements less the bins ordered plus the bins emptied today. In order to simulate the effect of different order windows, the number of days since the last order was placed is tracked. Each day whether an order is to be placed is calculated based on the number of days since the last order was placed and the maximum order frequency specified in the inputs. The maximum order frequency is equivalent to the inverse of the order cycle period. The order quantity is the number of bins to be ordered during that period; it is based on the order flag and the number of bins in the rolling bin requirement.

$$a_{x,42} = \text{Blank}$$

Column 42 is intentionally left blank to allow for future expansion of the model.

$$a_{x,43} = \text{Order Flag}$$
$$a_{x,44} = \text{Order Lead Time}$$

Whether an order is placed during a given period is calculated based on the Order Quantity in column 41, because even if it were time to place a scheduled order, if there were no bins to be replenished, an order would not be placed. Each day the order lead-time is calculated based on a normal distribution with an average and standard deviation given in the inputs and a random seed.

3.2.1.4 What is on order

$$a_{x,i+44} = \text{New order units due on the } i^{th} \text{ day in future}$$

Each day the number of bins ordered and the lead-time calculated for that order is used to populate columns 45-64 representing new orders due in the future.

$$a_{x,i+64} = \textit{Bins on order due i days into the future}$$

Each day the number of bins due is tracked, for each day up to the 20-day order window. The calculation to track future receipts is based on the previous days receipts adjusted for a new day and the newly placed order from that day.

$$a_{x,85} = \text{Blank}$$

Column 85 is intentionally left blank to allow for future expansion of the model.

3.2.2 Inputs

The inputs to the model comprise a 31-element vector used to capture all the variables on which the model calculations are based. The input vector captures both the variables expected to change between different samples of the experiment and the other values on which the model is based.

$$\begin{aligned} a_1 &= \text{Lead Time Seeds} \\ a_2 &= \text{Model 1 Seeds} \\ a_3 &= \text{Model 2 Seeds} \\ a_4 &= \text{Model 3 Seeds} \end{aligned}$$

The first four values are used to select which sets of random seeds to use for the calculations. For repeatability all of the random variables (uniformly distributed between zero and one) that are used in this model are stored in a large array with 10 unique sets for each purpose and samples to cover 5000 days. This allows the calculations to be repeated and compared with different inputs without the random number generation changing the results.

$$a_5 = \text{Number of Parts per Bin}$$
$$a_6 = \text{Number of Bins in System}$$

The key inventory controls in a kanban bin managed system are the number of parts per bin and the number of bins in the system. The multiple of these two values defines that maximum part capacity that can be held in inventory. The number of bins in the system is used as the primary lever of optimization; in more advanced generations of the model the number of bins is iteratively changed to find the best performing system.

$$a_7 = \text{Average Lead-Time}$$
$$a_8 = \text{Standard Deviation of Lead-Time}$$
$$a_9 = \text{Frequency of Orders}$$

The lead-time of replenishment is tracked as the bin empty to bin full time, encompassing both the material replenishment time and also any preparation time required before the parts can be used for production. This general definition of the lead-time can be used to model systems beyond just supplier replenishment but also earlier stages in a production environment. The standard deviation of the lead-time is also used to characterize the replenishment time for bins to be available. The frequency of orders is the maximum number of orders that can be placed during a 20-day order cycle.

$$a_{10} = \text{Model 1 Usage}$$
$$a_{11} = \text{Model 2 Usage}$$
$$a_{12} = \text{Model 3 Usage}$$

The model relies on the number of parts used per unit of production of each model to calculate the part requirements and consumption. The usages for each model can be set independently of the values used for other models.

a_{13} = Weekday Shift Length
a_{14} = Weekend Shift Length

The lengths of the shifts are used to calculate the amount of time available for production. Using two different inputs allows for adjusting the shifts independently.

a_{15} = Model 1 ProductionTime
a_{16} = Model 2 Produciton Time
a_{17} = Model 3 Production Time

The model relies on the production time in minutes per unit of production for each model to calculate the time requirements. The production time for each model can be set independently of the values used for other models.

a_{18} = Model 1 Production Average
a_{19} = Model 1 Production Standard Deviation
a_{20} = Model 2 Production Average
a_{21} = Model 2 Production Standard Deviation
a_{22} = Model 3 Production Average
a_{23} = Model 3 Production Standard Deviation

The model relies on the average and standard deviation of production based on historical numbers. For most of our production we have taken the average production to be the production schedule and based the standard deviation on the historical data.

a_{24} = Saturday Hurdle

The model calculates Saturday production shifts based on the hurdle rate. The hurdle rate represents the minimum number of back ordered units that are required to trigger Saturday production shift.

a_{26} = Part Value
a_{27} = Carry Rate
a_{28} = Order Cost
a_{29} = Model 1 Back Order Penalty
a_{30} = Model 2 Back Order Penalty
a_{31} = Model 3 Back Order Penalty

The final group of inputs is associated with the evaluation of the costs associated with each simulation run. The simulations are evaluated on the costs associated with carrying inventory, placing orders and compensating for back orders. To calculate the inventory holding costs, the part value and carry rate $\left(\frac{\$/year}{\$\ of\ inventory}\right)$ of inventory are multiplied times the average inventory across the days of the simulation. The number of orders is multiplied with the order cost to find the cost associated with placing orders. To find the back order penalty associated with each simulation run the number of each model back-ordered is multiplied with the backorder penalty per unit missed. These three costs represent the most important costs relevant to comparing the different runs of the simulation.

3.2.3 General Equations

Our model is built around a series of calculations that are used to fill in the matrix representing the state of the system for each day of the simulation model. The same calculations are used to fill out the matrix across each day and then continued down to the next day. The algorithm for filling out the matrix is explained in detail below, paired with a numerical example. The conditions for the numerical example are as follows:

20 parts/ bin
Mean lead-time = 5 days
Standard deviation of lead-time = 1 day
Ending Inventory (Day 1) = 170 units
Order placed for 1 bin
Lead-time = 1 day
Backordered units = 0
Order Frequency = Daily

Model Number	Usage (units)	Average Production (# tractors)	Std. Deviation of Production (# of tractors)	Production Cycle Time (minutes)
1	2	7	1	18
2	4	2	0.25	36
3	8	3	0.5	54

<u>Step 1:</u> The available inventory is calculated based on the previous day ending inventory and the orders scheduled to arrive that day.

E.g.: Available Inventory = Day 1 Ending Inventory + Day 2 Receipts
= 170 units + 1 bin x 20 parts/bin = 190 units

<u>Step 2:</u> The daily production schedule is calculated based on the specified random seeds and production characteristics. As discussed in the problem description our model assumes normally distributed production schedules with the average set at the nominal production level and the standard deviation set based on the historical production numbers.

E.g.: Via inverse of Normal distribution and randomly selected random seed,
Production(Model 1, Model 2, Model 3) = (8 units, 1 unit, 5 units)

Step 3: The daily rolling requirements are calculated based on the previous days requirements and the previous days production requirements. If scheduled production units are missed they are added to the rolling requirements to ensure they are built at the first opportunity.

E.g.: Required Production = Backordered units + Scheduled Units
Req'd Prod. Model 1 = 0 + 8 = 8
Req'd Prod. Model 2 = 0 + 1 = 1
Req'd Prod. Model 3 = 0 + 5 = 5

Step 4: The production part requirements are calculated based on the part usage per unit defined in the inputs and the rolling requirements for how many units of production are required.

E.g.: Parts Required = Req'd Prod. x Usage
Parts Req'd = 8 x 2 + 1 x 4 + 5 x 4 = 40 parts

Step 5: The "Part Availability Limited" production is calculated representing the maximum production based on the parts available for production. If there are not enough parts to produce the total rolling requirements, production is prioritized based on the model priority: Model 3, Model 2 then Model 1.

E.g.: Parts Req'd = 40 Parts available = 170;
$40 < 170 \rightarrow$ Part Limited = Required Production

Step 6: The total available production time is calculated based on weekday shift and if there is a Saturday shift. One of the assumptions we used in the model is that the production schedules for each tractor model are normally distributed and independent; this simplification can produce production schedules that are unrealistic and too long for the available production time. To accommodate this circumstance the available time in the shift is set to the longer of the time required to complete the production schedule and the nominal production shift.

E.g.: Production Time Required for Schedule = Req'd Prod. x Prod. Cycle Time
= 8 units Model 1 * 18 minutes + 1 unit Model 2 * 36 minutes + 5 units Model 3 * 54 minutes = 450 minutes

<u>Step 7:</u> The "Time Limited" production number is calculated based on the production time available during a shift and the "Part Availability Limited" production. If there is not enough time to complete the entire "Part Availability Limited" production the production is prioritized based on model priority: Model 3, Model 2 then Model 1.

E.g.: Production Time Required for Part-Limited = Production x Prod. Cycle Time
= 8 units Model 1 * 18 minutes + 1 unit Model 2 * 36 minutes + 5 units Model 3 * 54 minutes = 450 minutes
So, Time-Limited = Part-Limited

<u>Step 8:</u> The actual production numbers are calculated based on the "Time Limited" production.

E.g.: Model 1 = 8, Model 2 = 1, Model 3 = 5

<u>Step 9:</u> The part consumption is calculated based on the actual production and the part usages.

E.g.: Consumption = 40 Parts

<u>Step 10:</u> The ending inventory is calculated based on the starting inventory and the part consumption.

E.g.: Ending Inventory = Beginning Inventory – Consumption
= 170 units – 40 units = 130 units

Step 11: The number of "bins" of parts that are consumed are calculated based on the number of bins at least partially full at the start of the day and the number of bins at least partially full at the end of the day.

E.g.: Bins in Use Start = Beginning Inventory / Parts per Bin, rounded up
= 170 / 20, rounded up = 9 bins
Bins in Use End = Ending Inventory / Parts per Bin, rounded up
= 130 / 20, rounded up = 7 bins ➔ 2 bins emptied

Step 12: Each day a running tally of bins that have been emptied but not yet ordered are tracked based on the previous day's running tally, the previous day's number of bins ordered and the number of bins emptied today.

E.g.: Consumed Bins not yet Replenished, Day 1 = 0
Bins Emptied, Day 2 = 2 bins
Consumed Bins not yet Replenished, Day 2 = 2 bins

Step 13: Orders are placed based on the number of days since the last order and the specified maximum number of orders per 20-day order period from the inputs.

E.g.: Days since last order = 1 day;
Order Frequency = Daily ➔Order Flag = Yes ➔ Place Order

Step 14: The number of bins to order is set at the rolling quantity of bins that have been emptied but not yet ordered.

E.g.: Bins to be ordered = 2 bins

Step 15: The lead-time for each order is calculated based on the average and standard deviation of the lead-time in the inputs. We assume the lead-time to be normally distributed, and calculate the value for each day based on the normal probability density function and the random seed for that day. For the purposes of our model calculations the lead-time is rounded to the nearest whole number, and put at a ceiling of 20 and floor of 1 day. One day represents an order to be received the next day.

E.g.: Inverse of the Normal Distribution of Lead-Time (5, 1), given random seed =
= 3days

<u>Step 16:</u> Columns 45-84 are used to track the inbound orders. The first half is used to track the orders placed that day and the second half is used to track the incoming orders. Each day's orders are calculated based on the lead-time and order quantity that day, for days when orders are not placed the quantity will be zero and therefore not affect other calculations. Each day the incoming orders are added based on the previous days orders adjusted one day and the new orders placed that day. this paragraph is confusing.

E.g.: Column corresponding to 3 days out, update value to 2 bins

<u>Step 17:</u> The production losses are updated in columns 86-88. Our model doesn't track the actual serial numbers of the units being produced but we assume the production is first going to be that days scheduled units. For each tractor model we calculate the number of back-ordered units to be the production schedule for that model minus the number of units produced.. Logically the number of units back-ordered can't be less than zero therefore if production is more than the scheduled numbers due to completing previously back orders we set the back order number to zero.

E.g.: Back Orders = (Model 1 Prod. – Model 1 Sched.) + (Model 2 Prod. – Model 2 Sched.) + (Model 3 Prod. – Model 3 Sched.)
= (8 – 8) + (1 – 1) + (5 – 5) = 0

3.2.4 Summary Equations

For each simulation run of our model we used a standardized set of statistics to evaluate and compare the performance of the system under different entries. The two areas of summarization are the production statistics and the number of orders placed.

$$\sum_{i=1}^{n} \text{Daily Production Schedule}_i = \text{Total Schedule}$$

$$\sum_{i=1}^{n} \text{Daily Actual Production}_i = \text{Total Production}$$

$$\frac{\text{Total Production}}{\text{Total Schedule}} = \text{Raw Efficiency \%}$$

$$\sum_{i=1}^{n} \text{Daily Back Orderd Units}_i = \text{Missed First Time Through}$$

$$\frac{\text{Missed First Time Through}}{\text{Total Production}} = \text{First Time Through Efficiency \%}$$

These calculations are completed for each tractor model. To summarize the orders placed during each simulation the following equations are used:

$$\sum_{i=1}^{n} \text{Bins Ordered}_i = \text{Total Number of Bins Ordered}$$

$$\sum_{i=1}^{n} \text{Orders Placed}_i = \text{Total Orders Placed}$$

$$\frac{\text{Total Orders Placed}}{n} = \text{\% of Days When Orders Placed}$$

$$\frac{\text{Total Number of Bins Ordered}}{\text{Total Orders Placed}} = \text{Average Number of Bins Ordered}$$

$$\frac{\sum_{i=1}^{n} \text{Lead Time}_i}{n} = \text{Average Lead Time}$$

None of the summary statistics are used in the calculations other than orders placed to calculate the order cost component of total relevant costs. Our model calculates lead-times even if an order isn't placed in a particular period and the average lead-time statistic is used more to validate that the distribution is consistent with the input than it is for calculations in the model.

3.2.5 Objective Function

The objective of our investigation is to minimize the total cost of replenishment for each system. The components of this cost are the carrying cost of inventory, the order cost associated with placing orders for replenishment and the back-order cost penalty associated with units that were backordered due to not having enough parts to complete production. The carry cost is calculated by taking the average inventory level multiplied by the part value and the carry rate for the system. The order cost is calculated by multiplying the number of orders placed by the cost per order. Finally, the backorder cost is calculated by multiplying the number of units backordered per model on the day they are missed by the penalty associated with back-ordering that particular model; each unit backordered incurs a single one-time-only backorder charge.

THIS PAGE IS INTENTIONALLY LEFT BLANK.

4 Analysis

In this section, we discuss the process flow of our analysis following the completion of the development of our simulation model as depicted below in Figure 7. We begin by discussing the deployment of the simulation model to generate the first set of empirical experiments. We then demonstrate how we used the results of this first experiment to determine the parameters of our model which were in fact significant contributors to the total relevant cost of the system. Using these parameters, we generate a second set of empirical experiments with a focused set of variable parameters. We then explain how these results were used to refine a generalized predictive model. Finally, we discuss the application and performance of our predictive model on data supplied by our sponsor company. We performed the analysis in this order to avoid a situation of "model contamination" – where the same data is used to build and validate the model. By running case study data through an independently generated model, we better illustrate the predictive model's efficacy in a real-world situation.

Figure 7: Process flow for analysis

4.1 Simulation Model Deployment

The key tool we used to conduct our analysis was the model discussed in the Methods section. As the model allowed for rapid comparison of different production configurations, we were able to investigate the impact of various part characteristics on the performance of the overall system. We initially implemented the model in Microsoft Excel to allow for easy debugging and manipulation of the data; however, Excel limited our ability to time-efficiently complete runs due to the sheer quantity of computations associated with each individual simulation run. We, therefore, converted the model from Excel into Mathworks' MATLAB to complete the bulk of the calculations. For a particular set of inputs, the results from the Excel and MATLAB versions are the same; however, MATLAB allowed us to script the inputs and batch process the bulk of the calculations. A printout of the code from the MATLAB m-files is available to the reader in Exhibits 3-7 of the Appendix. We would suggest this method for future teams looking to conduct a similar type of analysis.

4.2 First Experiment Set

Our first set of experiments was designed to formulate a general understanding of how the different input parameters affect the output of the system. Our goal was to determine which factors were the most important candidates for a more thorough and detailed investigation. Before beginning the experiment, we devised a working list of input factors that satisfied two criteria: (1) our sponsor could reasonably provide reliable inputs for the factor, and (2) the factor provides intuitive potential to be a significant contributor to the experimental outcomes. The resulting list is as follows:

1. Parts per bin – the nominal quantity of units that can be fit into a bin; this is often a preset value due to standardized bin sizes and warehouse space constraints
2. Average lead-time – based on trailing information from the production database
3. Standard deviation of lead-time – based on historic information from the production database
4. Order frequency – the number of orders that take place in a standard 20-workday month (for example, a part that is ordered once per month receives a value of 1)

5. Part usage – the quantity of the part necessary to build the respective end-product models
    a. usage for model 1
    b. usage for model 2
    c. usage for model 3
6. Production schedule – the quantity of units to be built
    a. model 1
    b. model 2
    c. model 3
7. Part cost – the finished per-unit value of inventory

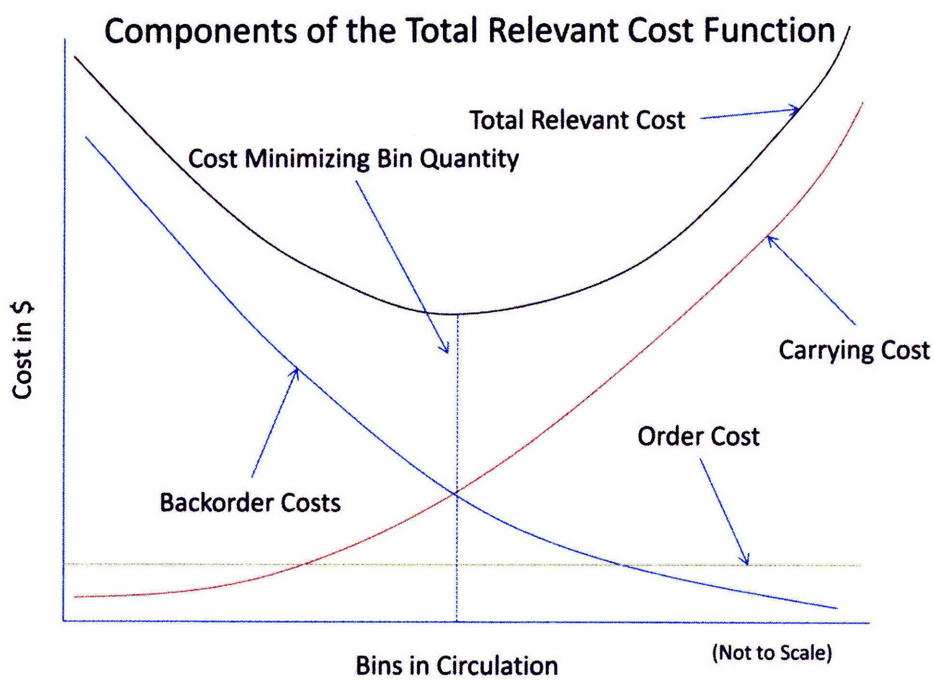
For each factor, we determined between 4 and 15 levels to investigate based on a reasonable step size resolution between the upper and lower bounds of the parameter's range. The bounds were based on a Pareto analysis of our sponsor's actual data for each of the listed parameters. Our goal was to design a full-factorial experiment to ensure the full independence of all test runs. However, the initial design indicated a permutation count on the order of 1,000,000,000. As such, it was not practical to complete a full-factorial experiment as originally planned given the computational time for each simulation run. We, therefore, condensed the experiment to eight parameters[3], selecting three levels for each representing the low, medium and high portions of the distributions. The new eight-factor-by-three-levels full-factorial comprised a total of 6,561(or 3^8) individual permutations. Because we suspected that the impacts between different characteristics would be very subtle, we avoided using some of the less robust experimental design techniques.

Each set of experiments consists of running a series of 5000-day simulations for each unique array of inputs. As the incremental use of random seed increases quantity of calculations per run by a factor of 4, we only had the resources to complete the

[3] Production schedule factors were eliminated

experiments with a single array of random seeds. For each unique array of inputs, we numerically determined the optimal number of bins using a step searching technique. The total relevant cost calculations were based on estimations of each component and may not exhibit absolute accuracy due to external factors that were not addressed by the model; nevertheless, they are consistent across various experimental runs. As such, the difference between total relevant costs for multiple experimental runs can be used for decision-making purposes. This total relevant cost is convex with respect to the number of bins, as pictured below in Figure 8. We used a step search to find the cost minimizing bin quantity, represented by the vertical line.

Figure 8: Components of the total relevant cost function. Note the cost minimizing quantity of bins denoted by the vertical line toward the center of the figure.



The output from each run of the simulation is an array summarizing the production outputs and costs associated with the individual run. The production outputs consist of total production by model and on-schedule production without backorders. The remainder of the array consists of performance statistics for the simulation run, such as average number bins ordered, total number of backorders by model and average inventory on hand. The output was formatted into a text file with each line representing a simulation run and the first columns representing the inputs and then the outputs.

To validate the optimization method of the model, a series of debugging trials were run on much smaller experiment sets. We ran the simulation model with different initial bin quantities to ensure the optimization would consistently find the same value. We also ran the same inputs in Excel and used the Solver numerical optimization tool to verify our answers. As a final check, we altered the optimal bin quantities generated by the model to verify that indeed the associated costs increased when we changed the number of bins used.

After completing the experimental runs, we conducted an analysis of variance (ANOVA) to evaluate the impact different factors had on both the optimal number of bins required and the costs associated with each scenario. This ANOVA was completed using statistical summaries generated in Excel, which can be found in Exhibits 1 & 2 in the Appendix. Upon investigation of the data, we did find one shortcoming of the model in that the optimization method did not accurately determine the optimal number of bins for parts with low parts-per-bin settings relative to usage. A side-effect of the optimization routine used to determine the required number of bins led to inconsistent results for one-part-per-bin systems because of the disproportionately large number of bins required to handle typical usage patterns. Specifically, model performance fell off when the number of bins in circulation exceeded 200; all such scenarios were one-part-per-bin systems. We were unable to determine modifications to the code that would consistently calculate the single part bin case. In practice, the only parts that fit a one-part-per-bin profile would likely be extremely large and high-value components for which a Kanban system would not be the proper inventory management technique. We ultimately removed experimental runs with parts-per-bin settings of 1 from the results and evaluated the remainder. The remaining factors and levels are summarized in Table 2.

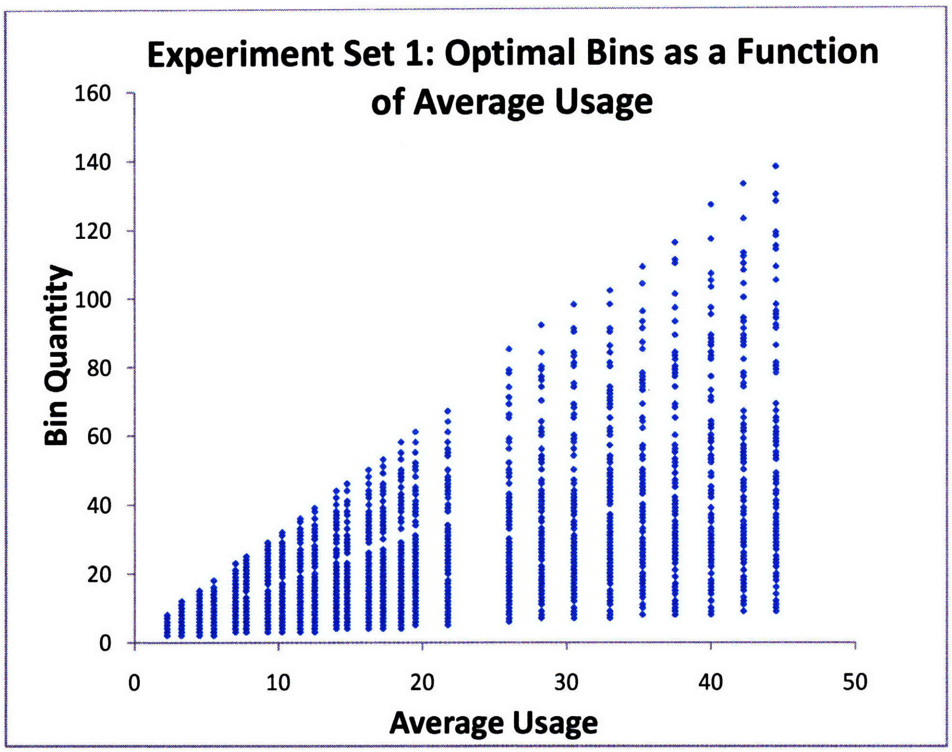
Table 2: Factors and levels for Experiment Set 1

Factor	**Levels**	**Units**
Parts/Bin	10, 20	qty/bin
Lead-Time	2, 3, 15	days
Standard Deviation of LT	.001, 1, 3	days
Order Frequency	2, 4, 10	orders/production month
Model 1 Usage	0, 1, 2	units per Model 1
Model 2 Usage	0, 1, 2	units per Model 2
Model 3 Usage	0, 1, 8	units per Model 3
Value	2, 16, 4096	$ value/unit

To determine which factors had a significant impact on the results of the simulation, we reviewed the results using a standard difference of means test. Those factor settings that showed a significant difference between their own average and the overall average Total Relevant Cost with a confidence level equal to $p < 0.05$ were designated as contributors to be escalated to the next round of experiments. This threshold provided an effective benchmark to eliminate the ineffectual factors from the experiment. Summary results can be found in Exhibits 1 and 2 of the Appendix.

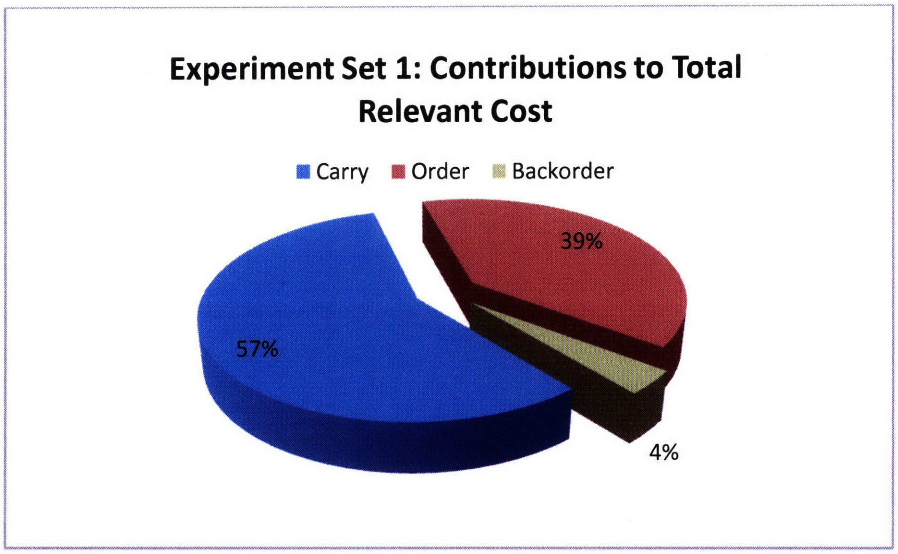
This led to deeper investigations of (1) the standard deviation of the lead-time, (2) order frequency, (3) usages and (4) part values. We further decided to trim part usages per model out of the next iteration since costs were intuitively and heavily correlated with the direct usage numbers. The graph below shows a distribution fanning from the origin, indicating that usage provides a linear contribution to the optimal bin quantity (see Figure 9). For each usage setting, the wide variability in the optimal bin quantity is attributable to other factors.

Figure 9: Bin Quantity vs. Average Part Usage, Experiment Set 1



Furthermore, the part usage per tractor was fixed from the viewpoint of the supply chain. Reviewing the costs of different experimental runs, we found that the order costs account for approximately 40% of total costs.

Figure 10: Components of total relevant cost and their contribution percentages



Because the order cost is a direct multiplicative result of the cost per order placed, we added the cost per order to the factors we would investigate in the next experiment.

4.3 Second Experiment Set

The goal of this set of experiments was to provide a much larger sample set of data that could be used to develop a predictive model capable of accurately determining the optimal outcomes of the simulation. We used the factors selected for further investigation from our first experiment set: standard deviation of lead-time, part value, order frequency and per order cost. We selected between five and eleven levels – again, based on an analysis of sponsor data – to investigate for each factor using a full-factorial design for the creation of individual runs. By using the simulation model as implemented in MATLAB, we were able to script the experiments and reduce the time and effort associated with completing such a large set of samples. Furthermore, this design reduced the complexity of interpreting the results because we did not have to investigate internal correlations between different levels. The factors and levels were implemented as described in Table 3.

Table 3: Factors and Levels for Experiment 2

Factor	Levels	Units
Standard Deviation of LT	0.001, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5	days
Part Value	4, 16, 32, 128, 512, 2048, 4096	$ value/unit
Order Frequency	1, 2, 4, 10, 20	orders per production month
Per Order Cost	0.1, 1, 2.5, 10, 20, 40, 75, 150, 300	variable cost with respect to the qty. of orders ($)

The other factors were set to representative levels for the system and held constant for all of the experimental runs. These settings are shown below in Table 4.

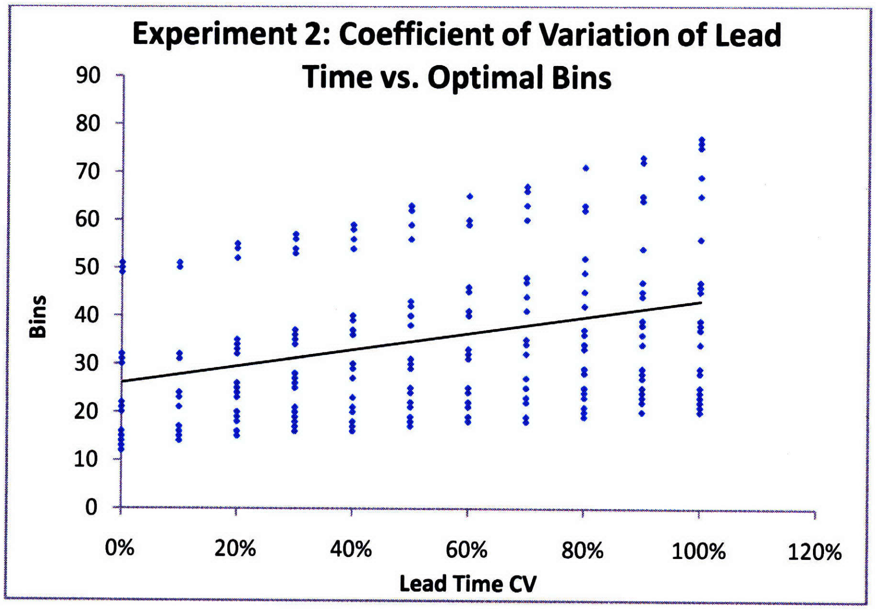
Table 4: Settings for remaining simulation model inputs

Lead Time	*Avg*	5	*days*
Part Usage	*Model 1 Usage*	2	*parts per tractor*
	Model 2 Usage	3	*parts per tractor*
	Model 3 Usage	6	*parts per tractor*
Production time Mintues	*Weekday*	430	*minutes*
	Sat	430	*minutes*
	Model 1 Per	18	*minutes*
	Model 2 Per	36	*minutes*
	Model 3 Per	54	*minutes*
Production Schedule	*Model 1 Avg*	7	*units per day*
	Model 1 Dev	1	*units*
	Model 2 Avg	2.25	*units per day*
	Model 2 Dev	0.5	*units*
	Model 3 Avg	3	*units per day*
	Model 3 Dev	0.75	*units*
	Sat Hurdle	5	*units*
	Days	5000	*days*
	Part Value	4	*$*
	Carry Rate	0.12	
BO Penalty	*Model 1*	2500	*$/unit backordered*
	Model 2	3500	*$/unit backordered*
	Model 3	5000	*$/unit backordered*

The results of each run with the optimally calculated number of bins were then analyzed via ANOVA to identify the general effects of each factor on the output. Upon review of the ANOVA results, we found the variability in the order cost to be directly related to the variability in order frequency and per order cost. The carry and backorder costs were correlated with changes in the standard deviation of the lead-time and the part value. The optimal number of bins and, therefore, the inventory buffer was a function of the standard deviation of the lead-time, the part value and the order frequency. In analyzing the results, we also determined that the number of bins had to be related to the average lead-time because it affects the amount of inventory float on order that had not been received. Based on our assumptions of inventory carrying cost being calculated based on the inventory on hand, the added float did not effect the relevant costs of the system; so, average lead-time was held constant for this set of experiments.

After a thorough review of the experimental results and ANOVA summary, we began evaluating the correlations between the inputs and results. We took the factor levels along with the average lead-time to be the inputs driving the results of the system. We used the optimal number of bins as the key result we wanted to predict along with the relevant costs associated with each set of inputs as secondary goals. We used xy-scatter plots to review how each of the factors related to the outputs. From the plots we were able to observe both the trend of that factor relative to the output as well as the level due to other factors. We used a least R-squared error line fit to quantify the trend of the output relative to the inputs. One of the most significant impacts we observed was of the coefficient of variation of the lead-time on the number of bins, as shown in Figure 11.

Figure 11: Optimal Bins as a function of CV of lead-time



Two observations from this chart are the increasing trend and the increasing spread. As the CV of lead-time increases, the number of bins necessary to compensate for the uncertainty in the system increases. Furthermore, increasing the CV of lead-time magnifies the variability of optimal bin quantities relative to the other factors. Based on our evaluation of the results coupled with our understanding of inventory policies and practices, we were satisfied with the simulation model's ability to accurately resemble real-world production. As such, the results of Experiment Set 2 would serve as the basis for development of our predictive model.

4.4 Predictive Model

The next phase of our analysis process entailed the development of a generalized predictive model that would accurately determine the number of bins found through our simulation. We began by going back to the basic inventory control equations and relating them to our system. The key goal of our project is the strategic deployment of inventory, which for our purposes is signified by the number of bins used to manage each part. More inventory means more parts and drives higher carrying costs, while less inventory might lead to more back orders. We define the purposes of the primary components of inventory as follows: *cycle stock* is used for the normal operations between two replenishment cycles, *in-transit stock* is used to account for inbound orders and *safety stock* is used to compensate for variability in the replenishment cycle. The cycle stock required to cover the period between orders can be directly calculated based on the time between orders, the part usage per unit of production and the average scheduled production. We show the explicit calculation for the cycle stock of item i, or C_i,here (measured in number of bins per order cycle):

$$C_i = \frac{Days/Order \ * \ Avg.Parts/Day}{Parts/Bin}$$

We show the explicit calculation for the in-transit stock of item i, or F_i, here (measured in number of bins):

$$T_i = \frac{C * Avg.LT * \frac{Orders}{Month}}{20 \ Days/Month}$$

We assume a 4-week, 5-day per week month; this is the origin of the "20 Days/Month" in the denominator of the equation for in-transit stock. Accounting for the average number of units on order is a function of the average order size, the length of the lead-time and the order frequency in orders per month.

Under the condition of zero variability in lead-time, we will designate the sum of those two stocks (i.e., cycle and in-transit) as the "perfect baseline":

$$Perfect\ Baseline_i = C_i + T_i$$

In a system with no variation in lead-time, the perfect baseline would be the optimal stocking level for the system. However, since very few supply chains have perfect delivery, a key to the strategic deployment of inventory is accounting for the variability. For many parts in our system the carrying cost of another bin of inventory is very small relative to the cost of back-ordering even a single unit; therefore, we focused on developing a very conservative buffer level. We converted the standard deviation of the lead-time from a number of days into the dimensionless coefficient of variation, CV_i, or the ratio of standard deviation to mean:

$$CV_i = \frac{\sigma_i}{\mu_i}$$

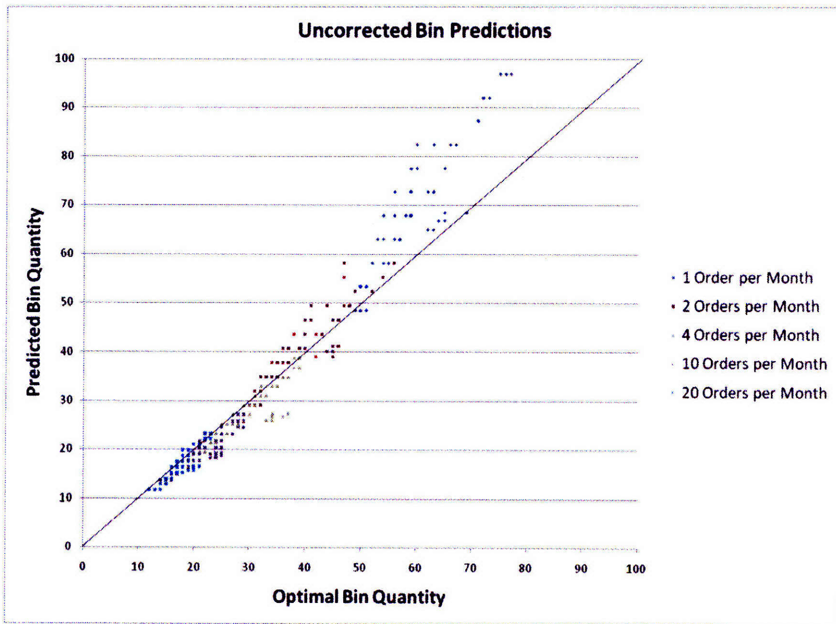
where σ_i is the standard deviation of lead-time for item *i* and μ_i is the average lead-time for item *i*.

To calculate the amount of inventory required for a real system (i.e. a system in which lead-time is variable), we take the perfect baseline and increase it by the lead-time coefficient of variation as a percentage of the perfect baseline. For example, if the average lead-time is five days and the standard deviation is two and a half days, the number of bins would be 50% higher than the perfect baseline. Hence the projected optimal number of bins is:

$$\mathit{Bin\ Prediction} = (C_i + T_i) * (1 + CV_i)$$

To evaluate this method of calculating the optimal number of bins, we used it to calculate the required number of bins for each of the runs from Experiment Set 2. We compared the optimal number of bins found in the simulation with the number of bins calculated using the predictive model. We used the percentage of error of our calculation relative to the optimal as the metric to evaluate the quality of our tool calculating the number of bins. The average absolute percent error of our tool was less than 35% for each run with the distribution approximately normal around zero. There were two modifications made to further refine our bin predictions: (1) rounding the number of bins to a whole number and (2) refining the equation through the introduction of a correction factor, $\beta_{correction}$. We plotted the percentage errors into a histogram, which showed that the bin prediction was slightly skewed to having too few bins; we chose to round up the number of bins required at the last step of each calculation because of this skewness combined with the relatively low cost of carrying extra bins. In order to improve our performance, we plotted the error percentage against different independent variables and did not find any particular factor that appeared to be driving the error; however, when we plotted the predicted number of bins against the error percentage, we did find that when we predicted fewer bins, we tended to under-predict and when we predicted more bins we tended to over predict, as shown below in Figure 12.

Figure 12: Uncorrected bin predictions



We used a least squares linear fit to find the level and trend of the residuals with an R-squared error of 63%, yielding our $\beta_{correction}$:

$$\beta_{correction,i} = \alpha + (C_i + T_i) * (1 + CV_i) * \beta$$

where α and β are numerically derived constants via linear regression. We then modified our prediction to compensate for the expected amount of error based on this linear fit:

$$\textit{Revised Bins Prediction} = \text{roundup}[(C_i + T_i) * (1 + CV_i) * (1 + \beta_{correction,i})]$$

Due to the clusters in the empirical predictions evident in Figure 12, we treated the Order Frequency as a discrete variable and generated pairs of α and β to correspond with each individual Frequency level via linear regression on the residuals. Our resulting α and β outcomes are shown below in Table 5:

Table 5: Frequency-based coefficients for the correction factor

Order Freq	Alpha	Beta
1 / month	.299	-.006
2 / month	.127	-.003
4 / month	.161	-.003
10 / month	.206	-.004
20 / month	.328	-.016

hence, yielding the final model:

$$\textit{Bins Prediction} = \text{roundup}[(C_i + T_i) * (1 + CV_i) * (1 + \beta_{correction,freq})]$$

$$\beta_{correction,freq} = \alpha_{freq} + (C_i + T_i) * (1 + CV_i) * \beta_{freq}$$

These modifications enabled us to further reduce the maximum absolute error percentage of empirical predictions to less than 25% with 80% of samples achieving absolute percent error less than 10%.

4.5 US Tractor Company Case Study

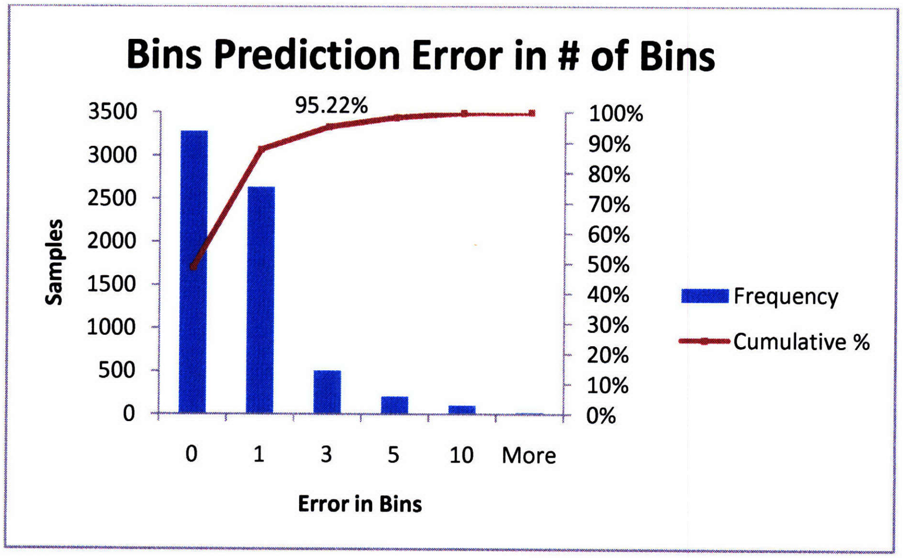
To evaluate the validity of the prediction model, we performed a case study based on parts from the US Tractor Company. We selected 557 parts from their inventory and used the production simulation to find the optimal number of bins for each part. We chose parts based on their per-unit usage, value and lead-time. We did not design the model to accommodate long lead-times believing the classic inventory calculation techniques could be more accurately used for those parts, since the lead-time coefficient of variation should be much lower. Per unit usages and values were selected to avoid the small value fasteners where optimal inventory policies are driven by other factors.

Because we did not have accurate data for per order cost and full usages across all models, ten different simulation runs were made varying the per order cost and part usages. We varied per order cost from $300 per order to $75 per order to cover what we expected to be a realistic range. To vary the part usages per unit of production we first completed runs at the usages we had data for and then we matched the per unit usage across the other models for which we didn't have information. We used our MATLAB model to simulate the production of each part and find the optimal number of bins for each one. The different scenarios we used for each part were:

- Usage exactly from data, order cost 300, parts/bin 20
- Under the condition that we did not have usage information for common parts, we assumed the usage in Models 1 and 2 to be the same as that in Model 3; the order cost and parts/bin were simulated as below:
  - Order Cost 300 Parts/Bin 20
  - Order Cost 150 Parts/Bin 20
  - Order Cost 75 Parts/Bin 20
  - Order Cost 300 Parts/Bin 50
  - Order Cost 150 Parts/Bin 50
  - Order Cost 75 Parts/Bin 50
  - Order Cost 300 Parts/Bin 100
  - Order Cost 150 Parts/Bin 100
  - Order Cost 75 Parts/Bin 100

After completing the simulation, we used the predictive model to calculate our expected number of bins so we could compare the results. The prediction equation we used was the same equation we developed based on the Experiment Set 2. We found that, based on our prediction, for 49% of over 5000 samples we correctly predicted the optimal number of bins. The prediction performance in terms of number of bins was within one bin of optimal for 88% of the samples and within 3 bins of optimal for 95% of the samples, as shown below in Figure 13.

Figure 13: Histogram and cumulative distribution of prediction errors



We further evaluated the results to define for which parts our prediction worked best. As stated above, small value parts may be managed and optimized for other considerations; our model does not accommodate them. The converse of small value parts are the very high value parts, which call for special attention in designing the system to balance the cost of carrying against the backorder cost of running out. Another constraint of our model is the length of lead-time. Due to the construction of the model, we limited the lead-time to twenty days. For parts that have a substantial likelihood of delivery beyond 20 days, the prediction's accuracy will decline. Furthermore, for bin predictions resulting in quantities in excess of 50 or less than 3, the model's performance loses reliability. Recognizing these limitations, we find the predictive model to be a valid tool for calculating the optimal number of bins.

THIS PAGE IS INTENTIONALLY LEFT BLANK.

5 Conclusions

The most important conclusion from our research is the connection between the coefficient of variation of the lead-time (days) and the amount of inventory that should be held. In developing areas for possible improvement of a system, the benefits of reducing the coefficient of variation of the lead-time may be more important than reducing the actual lead-time. Simply reducing the average lead-time while maintaining a high level of uncertainty in actual delivery times will not have that great an impact on the number of bins because while the perfect baseline decreases, the coefficient of variation by which it is multiplied increases. In order to improve system performance, both areas need to be addressed but particularly the variability in delivery; key characteristics of successful just-in-time systems are reliability and repeatability through the elimination of uncertainty.

5.1 Recommendations

This model for inventory deployment should be used as a baseline when selecting the number of bins to put into Kanban usage for a particular part number. However, the model was only designed to accommodate certain characteristics of the parts and any results from the model will have to be modified to accommodate other factors such as optimal order sizes, shipping economies, storage space and others.

5.2 Areas for Further Research

Our model was designed to investigate a limited number of the details about part usage and characteristics. Future research could be focused on expanding the part level detail used in determining the optimal number of parts by including the cost of warehousing or handling into the carrying cost rather than just the financing costs. Heavy tractors being a very stable product, our model does not handle any considerations of part spoilage or damage in storage that could increase with a higher level of inventory. Future models could also be adjusted to expand the time horizon of the model to a larger number of days into the future receipts or adjusting the resolution within a production day to accommodate multiple orders per production day. Particularly for high value parts, multiple daily or hourly deliveries might be more realistic to how the production system actually operates.

The end goal of just-in-time manufacturing systems is a consumption-based continuous flow of material through the system. As the bin size approaches one part per bin and single piece batch sizes, there are additional complexities that must be accommodated. Material handling costs between and within facilities may increase as the batch size gets smaller and smaller. Additional research could focus on the real costs associated with single part flow through the system, as well as quantifying the offset between the financial cost savings and the increased material handling costs. As the opportunities for improvement get smaller, the level of detail required to make good choices between different options gets more complicated. Calculating the true differences between two systems becomes a matter of accounting for all of the differences from ordering costs, to handling costs and administrative costs. One-piece flow systems will require high levels of coordination between suppliers and customers both internal and external.

Appendix

Exhibit 1: Experiment Set 1 Summary Results

		Parts/Bin	Lead-Time	StDev LT	Order Freq	Model 1	Model 2	Model 3	Value
Value			2	0.001	2	0	0	0	2
Bins	Average		16.73	20.11	30.55	16.96	22.79	12.85	24.35
	StDev		14.48	19.65	25.43	18.48	21.06	9.96	22.42
Carry	Average		535,617	399,243	734,124	404,476	526,052	301,281	960
	StDev		1,020,796	752,323	1,321,853	831,712	993,213	521,881	733
Order	Average		378,851	378,852	150,000	337,679	377,747	359,810	378,852
	StDev		242,524	242,526	0	213,843	240,694	231,760	242,526
BackOrder	Average		36,021	15,282	45,639	29,553	36,853	16,733	380
	StDev		91,521	41,014	113,565	77,972	88,568	42,878	5,126
TRC	Average		950,489	793,377	929,763	771,708	940,652	677,824	380,192
	StDev		1,101,742	784,754	1,420,545	916,970	1,074,452	579,345	242,113
Value		10	3	1	4	1	1	1	16
Bins	Average	31.22	18.30	23.09	22.80	22.88	22.89	15.21	24.35
	StDev	26.06	15.49	21.29	20.31	20.66	21.84	11.55	22.42
Carry	Average	542,281	547,459	523,800	531,406	533,900	532,132	360,403	7,679
	StDev	1,036,976	1,045,825	948,111	974,558	999,808	1,025,952	622,283	5,867
Order	Average	389,895	378,850	378,851	296,337	394,298	371,331	374,626	378,852
	StDev	247,120	242,523	242,525	19,613	249,060	241,577	237,362	242,526
BackOrder	Average	34,451	35,559	34,841	33,696	33,297	35,213	18,414	380
	StDev	85,413	92,366	74,933	78,170	86,846	87,884	46,902	5,126
TRC	Average	966,626	961,869	937,492	861,439	961,495	938,676	753,443	386,911
	StDev	1,111,600	1,125,734	1,021,975	1,042,727	1,072,385	1,107,107	670,963	241,034
Value		20	15	3	10	2	2	8	4096
Bins	Average	16.19	36.09	27.91	17.75	30.51	25.33	41.84	22.42
	StDev	13.02	27.61	23.94	17.34	23.92	22.69	25.73	20.87
Carry	Average	556,809	565,558	725,592	383,104	694,140	587,840	959,366	1,639,996
	StDev	1,044,255	1,055,170	1,315,456	700,291	1,214,430	1,094,325	1,497,053	1,210,231
Order	Average	367,805	378,849	378,846	690,213	399,998	387,349	399,998	378,846
	StDev	237,264	242,522	242,518	140,813	255,036	244,808	255,036	242,517
BackOrder	Average	37,224	35,931	57,388	28,177	43,963	35,559	70,241	106,751
	StDev	91,552	81,359	123,860	66,030	97,988	89,208	130,417	126,199
TRC	Average	961,837	980,338	1,161,826	1,101,494	1,138,101	1,010,747	1,429,606	2,125,592
	StDev	1,127,329	1,130,986	1,425,281	790,477	1,288,695	1,168,859	1,593,316	1,272,548
Bins	Average	23.70	23.70	23.70	23.70	23.70	23.70	23.70	23.70
	StDev	21.93	21.93	21.93	21.93	21.93	21.93	21.93	21.93
Carry	Average	549,545	549,545	549,545	549,545	549,545	549,545	549,545	549,545
	StDev	1,040,524	1,040,524	1,040,524	1,040,524	1,040,524	1,040,524	1,040,524	1,040,524
Order	Average	378,850	378,850	378,850	378,850	378,850	378,850	378,850	378,850
	StDev	242,465	242,465	242,465	242,465	242,465	242,465	242,465	242,465
BackOrder	Average	35,837	35,837	35,837	35,837	35,837	35,837	35,837	35,837
	StDev	88,536	88,536	88,536	88,536	88,536	88,536	88,536	88,536
TRC	Average	964,232	964,232	964,232	964,232	964,232	964,232	964,232	964,232
	StDev	1,119,362	1,119,362	1,119,362	1,119,362	1,119,362	1,119,362	1,119,362	1,119,362

Exhibit 2: Experiment Set 1 Summary Results, cont.

		Parts/Bin	Lead-Time	StDev LT	Order Freq	Model 1	Model 2	Model 3	Value
Value			2	0.001	2	0	0	0	2
Bins	Average		-29.43%	-15.16%	28.91%	-28.45%	-3.85%	-45.79%	2.72%
	StDev		-33.96%	-10.39%	15.98%	-15.74%	-3.95%	-54.59%	2.24%
Carry	Average		-2.53%	-27.35%	33.59%	-26.40%	-4.27%	-45.18%	-99.83%
	StDev		-1.90%	-27.70%	27.04%	-20.07%	-4.55%	-49.84%	-99.93%
Order	Average		0.00%	0.00%	-60.41%	-10.87%	-0.29%	-5.03%	0.00%
	StDev		0.02%	0.03%	-100.00%	-11.80%	-0.73%	-4.42%	0.03%
BackOrder	Average		0.51%	-57.36%	27.35%	-17.53%	2.83%	-53.31%	-98.94%
	StDev		3.37%	-53.68%	28.27%	-11.93%	0.04%	-51.57%	-94.21%
TRC	Average		-1.43%	-17.72%	-3.57%	-19.97%	-2.45%	-29.70%	-60.57%
	StDev		-1.57%	-29.89%	26.91%	-18.08%	-4.01%	-48.24%	-78.37%
Value		10	3	1	4	1	1	1	16
Bins	Average	31.72%	-22.81%	-2.60%	-3.81%	-3.45%	-3.44%	-35.83%	2.72%
	StDev	18.85%	-29.38%	-2.92%	-7.38%	-5.79%	-0.42%	-47.33%	2.24%
Carry	Average	-1.32%	-0.38%	-4.68%	-3.30%	-2.85%	-3.17%	-34.42%	-98.60%
	StDev	-0.34%	0.51%	-8.88%	-6.34%	-3.91%	-1.40%	-40.20%	-99.44%
Order	Average	2.92%	0.00%	0.00%	-21.78%	4.08%	-1.98%	-1.11%	0.00%
	StDev	1.92%	0.02%	0.02%	-91.91%	2.72%	-0.37%	-2.10%	0.03%
BackOrder	Average	-3.87%	-0.77%	-2.78%	-5.97%	-7.09%	-1.74%	-48.62%	-98.94%
	StDev	-3.53%	4.33%	-15.36%	-11.71%	-1.91%	-0.74%	-47.03%	-94.21%
TRC	Average	0.25%	-0.25%	-2.77%	-10.66%	-0.28%	-2.65%	-21.86%	-59.87%
	StDev	-0.69%	0.57%	-8.70%	-6.85%	-4.20%	-1.09%	-40.06%	-78.47%
Value		20	15	3	10	2	2	8	4096
Bins	Average	-31.72%	52.25%	17.76%	-25.10%	28.74%	6.86%	76.53%	-5.43%
	StDev	-40.61%	25.91%	9.15%	-20.91%	9.07%	3.45%	17.34%	-4.83%
Carry	Average	1.32%	2.91%	32.04%	-30.29%	26.31%	6.97%	74.57%	198.43%
	StDev	0.36%	1.41%	26.42%	-32.70%	16.71%	5.17%	43.87%	16.31%
Order	Average	-2.92%	0.00%	0.00%	82.19%	5.58%	2.24%	5.58%	0.00%
	StDev	-2.15%	0.02%	0.02%	-41.92%	5.18%	0.97%	5.18%	0.02%
BackOrder	Average	3.87%	0.26%	60.14%	-21.37%	22.67%	-0.78%	96.00%	197.88%
	StDev	3.41%	-8.11%	39.90%	-25.42%	10.68%	0.76%	47.30%	42.54%
TRC	Average	-0.25%	1.67%	20.49%	14.24%	18.03%	4.82%	48.26%	120.44%
	StDev	0.71%	1.04%	27.33%	-29.38%	15.13%	4.42%	42.34%	13.69%

* Cells with yellow fill denote statistically significant results as determined by a test of means at a confidence level of $p < 0.05$.

Exhibit 3, Run Experimental Trials

```
function []=runtrialfile()
%
%   Converting the trial running script into a function
%
%   no functional inputs or outputs
%   user interfacE:
%   input file:  the name of the file with the trials to run
%   output file: the name of the file in which to place the output
results
%
%   by Phil Hodge
%   for MLOG thesis 2008
%   Multi Echelon Inventory Control using Kanbans
%
%

clear all

%
% Get file names to use.
%

inputfile = input('what is the name of the input file?', 's');
%
% Input file format each:
%      input in delimited columns
%      trials on seperate lines
%

outputfile = input('where do you want the results put?', 's');
%
% Ourput file format each line represents a trial:
%      input array and then output array

%



%
% Initialize the Variables
%

seeds=load('randomseeds.txt');

trials=load(inputfile);

diminputs = size(trials);

numinputs = diminputs(1);

%
```

```
% Loop through to analyze set of inputs
%

for x = 1:numinputs,
    runoutputs(x,:)=optimizebins(trials(x,:),seeds, 0, -1);
    save(outputfile, 'runoutputs','-ascii');
    fclose('all');
     runcomplete = x
     totaltorun = numinputs

end
```

Exhibit 4, Begin Bin Search

```
function [optimizedresults]=optimizebins(input, seeds, fleetsize,
run)
%
% Function to find the optimal number of bins given the other input
% parameters.
%
% Input = vector of values similar to row 4 of excel defining system
% Output = optimized setting for parametrers and summary stats
%
% By Phil Hodge
% for MLOG 2008 Thesis
% Multi Echelon Inventory Control using Kanbans
%

%
% Guess the number of bins as a starting point
%

if fleetsize == 0

binguess=round(((input(7)+(20/input(9)))*((input(10)*input(18)+input
(11)*input(20)+...
      ...input(12)*input(22))))/input(5));
else
    binguess=round(fleetsize/input(5));
end


trial=input;
trial(6)=binguess;

trialstats=summstats(runtoplevelsim(trial,seeds),trial);

optimizedresults=seekbin(trial,trialstats,1,seeds, run);
```

Exhibit 5, Optimal Bin Search

```
function [results]=seekbin(input,runstats,step,seeds, run)
%
% Function to search for the number of bins to optimally run.
%
% Search is based on stepping forward with steps of 5 until TRC
stops
% inproving and then stepping back to find the best setting then
% steping forward again to verify number.
%
% The stepping is implemented using recursive calls of the same
function
% incrementing the number of bins until the total cost does not
improve
% then evaluating for other lines.
%
% by Phil Hodge
% for MLOG 2008 Thesis
% Multi Echelon Inventory Control Using Kanbans
%


trial=input;
trial(6)=trial(6)+step;

trialstats=summstats(runtoplevelsim(trial,seeds),trial);

if trialstats(28)<runstats(28)
    results=seekbin(trial,trialstats,step,seeds, run);
else
    if step == 5
        results = seekbin(input, runstats, -1, seeds, run);
    else
        if step == -1
            results = seekbin(input, runstats, 1, seeds, run);
        else
            results = [input runstats];
        end
    end
end
```

Exhibit 6, Summary Statistics

```
function [runstats]=summstats(runarray,inputvector)
%
% Function to summarize a run array representing the simulation of
% a set of imputs into the outputs.
%
% Input = array reprsentig the daily runs like the bulk of excel
% Output = vector representing the outputs.
%
% by Phil Hodge
% For MLOG Thesis 2008
% Multi Echelon Inventory Control using Kanbans
%
     runstats=zeros(1,28);
     runstats(1)=sum(runarray(:,6));
     runstats(7)=sum(runarray(:,7));
     runstats(13)=sum(runarray(:,8));
     runstats(2)=sum(runarray(:,30));
     runstats(8)=sum(runarray(:,31));
     runstats(14)=sum(runarray(:,32));
     runstats(3)=runstats( 1)-runstats( 2);
     runstats(9)=runstats( 7)-runstats( 8);
     runstats(15)=runstats( 13)-runstats( 14);
     runstats(4)=runstats( 2)/runstats( 1);
     runstats(10)=runstats( 8)/runstats( 7);
     runstats(16)=runstats( 14)/runstats( 13);
     runstats(5)=sum(runarray(:,86));
     runstats(11)=sum(runarray(:,87));
     runstats(17)=sum(runarray(:,88));
     runstats(6)=1-runstats( 5)/runstats( 1);
     runstats(12)=1-runstats( 11)/runstats( 7);
     runstats(18)=1-runstats( 17)/runstats( 13);
     runstats(20)=sum(runarray(:,41));
     runstats(22)=sum(runarray(:,43));
     runstats(21)=runstats( 22)/(max(runstats(:,1))-1);
     runstats(19)=runstats( 20)/runstats( 22);
     runstats(23)=mean(runarray(:,44));
     runstats(24)=mean(runarray(:,12));

runstats(25)=runstats(24)*inputvector(26)*inputvector(27)*(inputvect
or(25)/250);
     runstats(26)=runstats(22)*inputvector(28);

runstats(27)=runstats(5)*inputvector(29)+runstats(11)*inputvector(30
)+ runstats(17)*inputvector(31);
     runstats(28)=runstats(25)+runstats(26)+runstats(27);
```

Exhibit 7, Simulation Code

```
function [calcs]=runtoplevelsim(inputvector, seeds)
%
% Function to Run Bulk of calculations for a given
% set of inputs.
%
% Input = vector of values similar to row 4 of excel
% Output = giant matrix in rows 33 - 5032 of excel

%
% Summarization to be done by another function
%
% By Phil Hodge
% for MLOG 2008 Thesis
% Multi Echelon Inventory Control using Kanbans
%

days=inputvector(25);

rand1=inputvector(1);
rand2=inputvector(2)+10;
rand3=inputvector(3)+20;
rand4=inputvector(4)+30;

calcs=zeros(days+1,88);
calcs(1,12)=inputvector(5)*inputvector(6);
calcs(1,34)=calcs(1,12);



%
% Loop through to fill out the matrix of values.
%

for x=2:days+1,
    calcs(x,1)=x-1; % Day being Simulated
    calcs(x,2)=seeds(x-1,rand1); % seed value for leadtime
    %
    % Scheduled production numbers by model
    %
    calcs(x,6)=round(norminv(seeds(x-
1,rand2),inputvector(18),inputvector(19)));
    calcs(x,7)=round(norminv(seeds(x-
1,rand3),inputvector(20),inputvector(21)));
    calcs(x,8)=round(norminv(seeds(x-
1,rand4),inputvector(22),inputvector(23)));
    %
    % Required production numbers by model based on schedule and
    % backorders
    %
    calcs(x,9)=calcs(x,6)+calcs(x-1,9)-calcs(x-1,30);
    calcs(x,10)=calcs(x,7)+calcs(x-1,10)-calcs(x-1,31);
    calcs(x,11)=calcs(x,8)+calcs(x-1,11)-calcs(x-1,32);
```

```
    %
    % Inventory available for production
    %
    calcs(x,12)=calcs(x-1,34)+calcs(x-1,65)*inputvector(5);
    %
    % Part requirements based on prduction requirements both total
across
    % models and by model
    %
    calcs(x,14)=calcs(x,9)*inputvector(10);
    calcs(x,15)=calcs(x,10)*inputvector(11);
    calcs(x,16)=calcs(x,11)*inputvector(12);
    calcs(x,13)=calcs(x,14)+calcs(x,15)+calcs(x,16);
    %
    % Calculation of part limited production based on the availible
parts
    % and production requirements
    %
    if calcs(x,13)<calcs(x,12)
            calcs(x,18)=calcs(x,9);
            calcs(x,19)=calcs(x,10);
            calcs(x,20)=calcs(x,11);
    else
            if calcs(x,16)<calcs(x,12)
                calcs(x,20)=calcs(x,11);
                if calcs(x,15)<=(calcs(x,12)-calcs(x,16))
                    calcs(x,19)=calcs(x,10);

calcs(x,18)=min(calcs(x,9),max(0,floor((calcs(x,12)-calcs(x,19)*...
                       ...inputvector(11)-
calcs(x,20)*inputvector(12))/inputvector(10))));
                else
calcs(x,19)=min(calcs(x,10),max(0,floor((calcs(x,12)-calcs(x,20)*...
                 ...inputvector(12))/inputvector(11))));

calcs(x,18)=min(calcs(x,9),max(0,floor((calcs(x,12)-calcs(x,19)*...
                       ...inputvector(11)-
calcs(x,20)*inputvector(12))/inputvector(10))));
                end
            else
calcs(x,20)=min(calcs(x,11),max(0,floor(calcs(x,12)/inputvector(12))
));

calcs(x,19)=min(calcs(x,10),max(0,floor((calcs(x,12)-calcs(x,20)*...
                 ...inputvector(12))/inputvector(11))));
                calcs(x,18)=min(calcs(x,9),max(0,floor((calcs(x,12)-
calcs(x,19)*...
                 ...inputvector(11)-
calcs(x,20)*inputvector(12))/inputvector(10))));
            end
    end
    %
```

```
    % Fill in production requirements as full if the usage for a
part is
    % zero for a particular model.
    %
    if inputvector(10)==0
        calcs(x,18)=calcs(x,9);
    end
    if inputvector(11)==0
        calcs(x,19)=calcs(x,10);
    end
    if inputvector(12)==0
        calcs(x,20)=calcs(x,11);
    end

    if (mod(calcs(x,1),5)==0) &&
((calcs(x,9)+calcs(x,10)+calcs(x,11)-calcs(x,6)-calcs(x,7)-...
           ...calcs(x,8))>inputvector(24))
        calcs(x,21)=1;
    else
    end
    %
    % Calculation of the time limited production
    %

calcs(x,22)=max(inputvector(13),calcs(x,6)*inputvector(15)+calcs(x,7
)*inputvector(16)+...
      ...calcs(x,8)*inputvector(17))+inputvector(14)*calcs(x,21);
    calcs(x,24)=calcs(x,18)*inputvector(15);
    calcs(x,25)=calcs(x,19)*inputvector(16);
    calcs(x,26)=calcs(x,20)*inputvector(17);
    calcs(x,23)=calcs(x,24)+calcs(x,25)+calcs(x,26);

    if calcs(x,23)<calcs(x,22)
            calcs(x,27)=calcs(x,18);
            calcs(x,28)=calcs(x,19);
            calcs(x,29)=calcs(x,20);
    else
            if calcs(x,26)<calcs(x,22)
                calcs(x,29)=calcs(x,20);
                if calcs(x,25)<(calcs(x,22)-calcs(x,26))
                    calcs(x,28)=calcs(x,19);

calcs(x,27)=min(calcs(x,18),max(0,floor((calcs(x,22)-calcs(x,28)*...
                       ...inputvector(16)-
calcs(x,29)*inputvector(17))/inputvector(15))));
                else
calcs(x,28)=min(calcs(x,19),max(0,floor((calcs(x,22)-calcs(x,29)*...
                       ...inputvector(17))/inputvector(16))));

calcs(x,27)=min(calcs(x,18),max(0,floor((calcs(x,22)-calcs(x,28)*...
                       ...inputvector(16)-
calcs(x,29)*inputvector(17))/inputvector(15))));
                end
```

```
            else
calcs(x,29)=min(calcs(x,20),max(0,floor(calcs(x,22)/inputvector(17))
));

calcs(x,28)=min(calcs(x,19),max(0,floor((calcs(x,22)-calcs(x,29)*...
                 ...inputvector(17))/inputvector(16))));

calcs(x,27)=min(calcs(x,18),max(0,floor((calcs(x,22)-calcs(x,28)*...
                 ...inputvector(16)-
calcs(x,29)*inputvector(17))/inputvector(15))));
            end
    end
    %
    % Actual prodction of each model based on time limited
calculations
    %
    calcs(x,30)=calcs(x,27);
    calcs(x,31)=calcs(x,28);
    calcs(x,32)=calcs(x,29);
    %
    % Calculate part consumption and ending inventory
    %

calcs(x,33)=calcs(x,30)*inputvector(10)+calcs(x,31)*inputvector(11)+
calcs(x,32)*inputvector(12);
    calcs(x,34)=calcs(x,12)-calcs(x,33);
    %
    % Calculate the number of bins emptied this day
    %
    calcs(x,35)=ceil(calcs(x,12)/inputvector(5));
    calcs(x,36)=ceil(calcs(x,34)/inputvector(5));
    calcs(x,37)=calcs(x,35)-calcs(x,36);
    calcs(x,38)=calcs(x,37)+calcs(x-1,38)-calcs(x-1,41);
    %
    % Order flag to calculate if an order gets produced this day and
how
    % many bins to order if an order is placed.
    %
    if (calcs(x-1,39)+1)>=(20/inputvector(9))
       calcs(x,40)=1;
    else
    end
    if(calcs(x,40)==1)
        calcs(x,39)=0;
    else calcs(x,39)=calcs(x-1,39)+1;
    end
    calcs(x,41)=calcs(x,40)*calcs(x,38);
    if calcs(x,41)>0
        calcs(x,43)=calcs(x,40);
    else calcs(x,43)=0;
    end
    calcs(x,44)=max(1,min(20,round(norminv(seeds(x-
1,rand1),inputvector(7),inputvector(8)))));
```

```
    %
    % Track inbound orders
    %
    scratch = 44+calcs(x,44);
    calcs(x,scratch)=calcs(x,41);

    for b=1:20,
        calcs(x,b+64)=calcs(x-1,b+65)+calcs(x,b+44);
    end
    %
    % Calculate the back orders for each model.
    %
    calcs(x,86)=max(0,calcs(x,6)-calcs(x,30));
    calcs(x,87)=max(0,calcs(x,7)-calcs(x,31));
    calcs(x,88)=max(0,calcs(x,8)-calcs(x,32));

end
```

Biographical References

Phil Hodge is a candidate for an MIT Master of Engineering in Logistics. Prior to returning to the Boston area, Phil most recently worked in Dearborn, Michigan as a Manufacturing and Material Flow Engineer for Ford Motor Company. Phil's major focus at Ford was launching major vehicle programs. He worked on both the 2004 F-150 and 2008 Super Duty that make up the Ford F-Series line, the most popular line of vehicles in the United States. Following completion of his degree, Phil will be joining PRTM Management Consultants in the Automotive, Aerospace and Industrial Practice. Phil holds an M.S. in Manufacturing Engineering from Boston University in Boston, Massachusetts and a B.S. in Mechanical and Aerospace Engineering from Cornell University in Ithaca, New York.

Josh Lemaitre is a candidate for an MIT Master of Engineering in Logistics. Prior to returning to the Boston area, Josh most recently worked in Salt Lake City, Utah, as an Internal Consultant in the Merchandising Operations division of Overstock.com, a discount online retailer. He was responsible for sales forecasting, yield management, marketing analytics and the development of the company's personalization program. Following graduation, Josh will perform the role of Director of Analytics & Product for richrelevance, a San Francisco based Web 2.0 startup focused on the space of personalization technology. Josh holds a B.S. in Economics from Duke University in Durham, North Carolina.

Bibliography

Abhyankar, H. S., and Graves, S. C. (2000, January). Creating an Inventory Hedge for Markov-Modulated Poisson Demand: An Application and Model. Cambridge, MA.

Aviv, Y., and Federfruen, A. (2001). Capacitated Multi-Item Inventory Systems with Random and Seasonally Fluctuating Demands: Implications for Postponement Strategies. *Management Science*, *47* (4), 512-531.

Bollapragada, R., Rao, U. S., and Zhang, J. (2004). Managing Inventory and Supply Performance in Assembly Systems with Random Supply Capacity and Demand. *Management Science, 50(12)*, 1729-1743.

Bookbinder, J. H., and Heath, D. B. (1988). Replenishment Analysis in Distribution Requirements Planning. *Decision Sciences*, *19* (3), 477-489.

Buzacott, J. A., and Shanthikumar, J. G. (1994). Safety Stock versus Safety Time in MRP Controlled Production Systems. *Management Science, 40(12)*, 1678-1689.

Chopra, S., and Lariviere, M. A. (2005). Managing Service Inventory to Improve Performance. *MIT Sloan Management Review*, *47* (1), 55-63.

Chopra, S., Reinhardt, G., and Dada, M. (2004). The Effect of Lead-time Uncertainty on Safety Stocks. *Decision Sciences*, *35* (1).

Clark, A. J., and Scarf, H. (1960). Optimal Policies for a Multi-Echelon Inventory Problem. *Management Science*, *6*, 475-490.

Deleersnyder, J.-L., Hodgson, T. J., Muller, H., and O'Grady, P. J. (1989). Kanban Controlled Pull Systems: An Analytic Approach. *Management Science, 35(9)*, 1079-1091.

Diks, E. B., and de Kok, A. G. (1999). Computational results for the control of a divergent N-echelon inventory system. *International Journal of Production Economics*, *59*, 327-336.

Ebrahimpour, M., and Fathi, B. M. Dynamic Simulation of a Kanban Production Inventory System. *International Journal of Production Management*, *5* (1), 5-14.

Esparrago, R. A. (1988). KANBAN. *Production and Inventory Management Journal*, *29* (1), 6-10.

Graves, S. C. (1987, June). Safety Stocks in Manufacturing Systems. Cambridge, MA.

Graves, S. C., and Willems, S. T. (2000). Optimizing Strategic Safety Stock Placement in Supply Chains. *Manufacturing and Service Operations Management*, *2* (1), 68-83.

Gumus, A. T., and Guneri, A. F. (2007). Multi-echelon inventory management in supply chains with uncertain demand and lead-times: literature review from an operational research perspective. *Journal of Engineering Manufacture*, *221*, 1553-1570.

Gupta, S., and Brennan, L. (1993). A Knowledge Based System for Combined Just-In-Time and Material Requirements Planning. *Computers Elect. Engng. Vol. 19, No.2*, 157-174.

Gurgur, C. Z., and Tayfur, A. (2004). Approximate Analysis of Decentralized, Multi-Stage Pull-Type Production/Inventory Systems. *Annals of Operations Research*, *125*, 95-116.

Inderfurth, K. (1991). Safety stock optimization in multi-stage inventory systems. *International Journal of Production Economics*, *24*, 103-113.

Jothishankar, M. C., and Wang, H. P. (1992). Metamodelling a Just-in-Time Kanban System. *International Journal of Production Management*, *13* (8), 18-36.

Kochel, P., and Nielander, U. (2002). Kanban Optimization by Simulation and Evolution. *Production Planning and Control*, *13* (8), 725-734.

Lagodimos, A. G., and Anderson, E. J. (1993). Optimal Positioning of Safety Stocks in MRP. *International Journal of Production Research*, *31* (8), 1797-1813.

Lee, H. L., and Billington, C. (1993). Material Management in Decentralized Supply Chains. *Operations Research*, *41* (5), 835-847.

Minner, S. (2000). *Strategic Safety Stocks in Supply Chains.* Berlin; New York: Springer.

Mitra, S., and Chatterjee, A. K. (2004). Leveraging information in multi-echelon inventory systems. *European Journal of Operational Research*, *152*, 263-280.

Nori, V. S., and Sarker, B. R. (1998). Optimum number of Kanbans between two adjacent stations. *Production Planning and Control*, *9* (1), 60-65.

Philipoom, P. R., Rees, L. P., Taylor, I. B., and Huang, P. Y. (1987). An investigation of the factors influencing the number of Kanbans required in the implementation of the JIT technique with Kanbans. *International Journal of Production Research*, *25* (3), 457-472.

Quintana, R., and Leung, M. T. (2007). Adaptive exponential smoothing versus conventional approaches for lumpy demand forecasting: case of production planning for a manufacturing line. *Internation Journal of Production Research*, 4937-4957.

Ronen, D. (1990). Inventory Centralization/Decentralization - The "Square Root Law" Revisited Again. *Journal of Business Logistics*, *11* (2), 129-138.

Rosling, K. (1989). Optimal Inventory Policies for Assembly Systems Under Random Demands. *Operations Research*, *37* (4), 565-579.

Sengupta, S., Sharief, F., and Dutta, S. P. (1999). Determination of the Optimal Number of Kanbans and Kanban Allocation in a FMS: a simulation-based study. *Production, Planning and Control*, *10* (5), 439-447.

Singh, N., and Shek, K. H. (1989). The Development of a Kanban System: a Case Study. *International Journal of Production Management*, *10* (7), 28-36.

Stenger, A. J. (1996). Reducing inventories in a multi-echelon manufacturing firm - A case study. *International Journal of Production Economics*, *45*, 239-249.

Takahashi, K. (2003). Comparing reactive Kanban systems. *International Journal of Production Research*, *41* (18), 4317-4337.

Takahashi, K., and Nakamura, N. (2002). Decentralized reactive Kanban system. *European Journal of Operational Research*, *139*, 262-276.

Tee, Y.-S., and Rossetti, M. D. (2002). A robustness study of a multi-echelon inventory model via simulation. *International Journal of Production Economics,*, *80*, 265-277.

Wang, S., and Sarker, B. R. (2005). An assembly-type supply chain system controlled by kanbans under a just-in-time delivery policy. *European Journal of Operational Research*, *162*, 153-172.

Wang, S., and Sarker, B. R. (2006). Optimal models for a multi-stage supply chain system controlled by kanban under just-in-time philosophy. *European Journal of Operational Research, 172*, 179-200.

Xiaobo, Z., Gong, Q., and Wang, J. (2002). On control strategies in a multi-stage production system. *International Journal of Production Research*, *40* (5), 1155-1171.

Zaenglein Jr., R. (2000). External Kanban Systems in Automotive Assembly. Cambridge, Massachusetts.