

A Comparative Paper on Four Spare Parts Management Methodologies

10/19/95

THE AVAILABILITY MAXIMIZER MODEL

10/23/95

The Development Environment of the Availability Maximizer

An important part of understanding the Availability Max lies in understanding its development environment and why the client required the particular type of inventory system XLTS co-developed. The Availability Max was written expressly for a low demand environment at the dealer level. Most independent demand inventory algorithms (EOQ, POQ, Silver-Meal, Parts-Period Balancing, etc.) rely on the statistics of high demand per replenishment period to determine inventory holding and purchasing amounts. For environments with relatively stable demand these methods work reasonably well. Spare parts demand, in contrast, is generally both low in demand per part with significant variability from period to period. As we will find later, low demand and demand variability are related characteristics. A second limiting factor is that while the spare part demand is low per item, the average spare parts depot will have to carry many times more parts which must be carried than a manufacturing depot to generate comparable fill levels. Spare parts operations must carry not only this model year's inventories, but inventories going back decades. These three characteristics of the spare parts environment: low demand, highly erratic demand and massive parts databases present difficult challenges to a company dedicated to order fulfillment.

A second feature of the client's environment was the lack of ownership of the dealership network by the client. In fact many of the dealers sold competing agricultural-construction equipment brands and maintained spare parts for these brands in their stockrooms in addition to CASE spare parts. Because the dealers would not allow CASE to manage their inventories with the Availability Max model without experiencing significant benefits, any system used would have to both increase fill and reduce the inventory carry amounts.

From the environmental challenges described above, it should be clear that the client needed a inventory system much different in order to address the characteristics specific to its part business. XLTS Consulting concluded that the Availability Max was tool that fit the environmental requirements.

The Basic Part Selection Logic of the Availability Maximizer

The entire logic within the Availability Maximizer is based on two simple inventory goals.

1. *The limited resources of:*
 - a. *Capital required to carry inventory over order intervals*
 - b. *Physical space in the stockroom at the dealer*
2. *The company's interest in filling as many customer orders as possible*

These two inventory goals are incorporated into the Availability Max as the following:

1. *The Cost of the Part (limited resources)*
2. *Expected Additional Demand Satisfied (companies interest in filling customer orders)*

Why Cost of the Part?

As was noted in the first section this paper, due to the low demand per part, the erratic nature of demand, and the vast numbers of parts in a typical spare parts database a depot would have to carry many times as much inventory as a manufacturing operation of the same general size in order to generate a comparable order fill. Because this would be so expensive, the typical spare parts operation must accept a certain level of stockout on some parts, and a 100% stockout on the lowest demand parts in its database. Even after significant inventory intelligence has been used, after some point, the fill level is based upon the aggregate inventory dollars the dealer is willing to commit to order fulfillment. At some point, the customer is no longer willing to subsidize higher fill levels with higher part prices. Therefore inventory dollars investment is a key component order fill. Inventory investment composed of the aggregate of all parts in inventory. The inventory system can either buy more less expensive parts or fewer more expensive parts. Therefore, in the algorithm in the Availability Max, the cost of the part is the denominator to the objective function which Availability Max is attempting to maximize for each part.

Why the Expected Additional Demand Satisfied?

Consider yourself in the situation of the parts manager at a CASE dealership. Every week you must make a decision as to which parts to order. You, presumably, want to order parts which will sell quickly, which would mean your new purchases would take up less space on your shelves, free up monies for further purchases, and please more customers. But, which parts are the best parts to order. You could simply purchase whatever you sold the past week and that would get you part of the way there. Or, you could analyze the past year's demand and with statistical methods determine the probability of demand on different parts. This analysis would

yield the Expected Additional Demand Satisfied given a certain order amount. The EADS will always be smaller, or in rare instances the same as amount that you chose to order. As it is impossible to satisfy demand for parts you do not have, EADS will never be bigger than the order amount. The calculation of the EADS is very simple. The current inventory position is compared to the yearly demand of the part in order to determine the probability of a additional demand in some multiple of the order size. If the beginning inventory position is small in relation to the lead time demand, then there is a probabalistically larger chance of unfulfilled demand than if the beginning inventory position were larger than the lead time demand. (*later in this paper the specifics of the probability distributions used and their calculations will be expanded upon*). Remember that the second basic objectives for which the model is built is the *companies interest in filling as many customer orders as possible*. EADS is simply a

Basic Rule of The Incremental Benefit of Ordering Order Amount (Q)

$$EADS = \% \text{ of } Q$$

Objective Function

The objective function is where the two mathematical expressions of the two inventory goals are put to use. The objective function is the goal that is to be optimized by the Availability Max. In this case we want the objective function to be maximized. This will allow the model to select parts which have a high EADS in relation to their cost.

Objective function :

$$\text{Maximize } (Expected \text{ Additional Demand Satisfied}) / (\text{unit cost})$$

Determining the numerator, the EADS, to the objective function is where the majority of effort in the Availability Max model is expended. The relative cost of a part compared to its probability of being subject to a customer demand determines it's ranking as either a high or low opportunity part.^{1 2} For two equivalently priced parts, the higher opportunity part is the one with

¹ In practice, since there is neither a strong correlation between expensive parts nor cheap parts with demand history, the more expensive parts are at a disadvantage and typically the last to be purchased by the Availability Max. The degree to which it purchases mostly medium to less expensive depends upon the desired overall service level used as an input. The higher the desired service level the higher the model will purchase on the cost scale.

² The model is technically defined as an optimizer. This is because it iteratively compares each and every part until it finds the optimum combination given the objective function, or until it hits a

the highest ENDS as a percentage of its order amount. The Availability Max performs the objective function above iteratively. This means that, beginning from a current inventory position, it calculates the objective function for every part in the parts order amount as many times as is necessary. After a single iteration where high opportunity parts are selected for purchase, the purchase amount is then added to current inventory and the objective function is calculated again. For the parts purchased on the previous iteration, their opportunity is reduced to reflect the new higher stocking position of those items. It is important to remember that no part will be identified as a high opportunity part for all model calculations. At some point the sufficient inventory is purchased through prior iterations that the part no longer an attractive alternative to add more inventory. To provide perspective, for the average dealer used in the development of this model, it is common for the model to perform 7000 iterations for purchases and returns before arriving at the optimum holding position.

Example 1 shows how the model would choose the parts at different iterations with the demand and cost characteristics in Table 1 .

Example. 1

Below are the demand probabilities and costs for part A and part B:

TABLE 1

	Demand of 1	Demand of 2	Demand of 3	Part Cost
Part A	.4	.2	.1	\$5
Part B	.6	.1	.05	\$8

First Iteration .4/5 > .6/8, carry 1 of Part A

Second Iteration .2/5 < .6/8, carry 1 of Part B

Third Iteration .2/5 > .1/8, carry another of Part A

Final total after three iterations: carry 2 of A and 1 of B

In the first iteration, the probability of a Demand of 1 of Part A divided by Part Cost A. This is compared against Demand of 1 of Part B divided by Part Cost B. However, notice that after the first iteration, the relevant question becomes the probability of a Demand of 2 on Part A

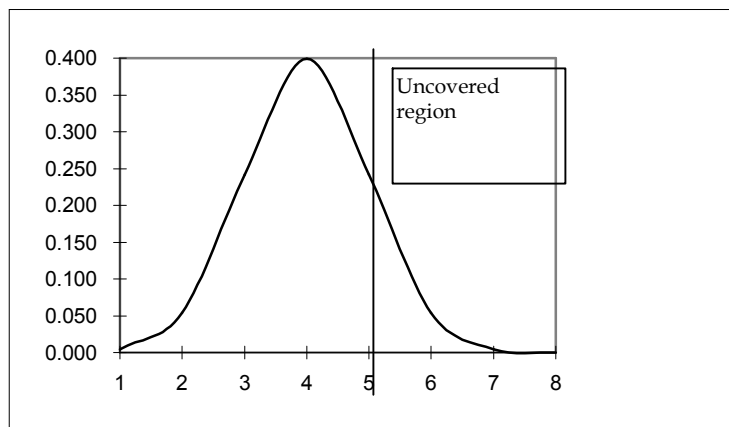
constraint. The constraints are set by the user and include, total inventory dollars, new inventory purchases, individual service level, global service level, and iteration cap.

vs. a probability of a Demand of 1 for Part B. This is because 1 of Part A has already been purchased.³ Therefore, the Availability Max model asks, “What is the incremental probability of moving from 1 to 2 demand of Part A vs. the probability of a demand of moving from 0 to 1 demand of Part B.” It is important not to skim over the past paragraph as it is the basic operating logic of the model.

The Specifics of the Objective Function for Part Purchases

As explained above, the EADS is generated by comparing the current inventory position to the demand of the past year. Given a certain level of demand, and a certain level of inventory, there is a section under the curve which is left uncovered by the current inventory holding position. Graph 1 displays a situation with a lead time demand of 4 units, and a average inventory of 5 units. Clearly, demands of 6, 7, 8 units and above would be stocked-out 1 unit (6-5), 2 units, (7-5), and 3 units (8-5) respectively. Any demand up to 5 will be covered by the current inventory.

*Graph 1
Probabilities of Demands Above the Inventory Level 5*



The logic of the model as it was presented to XLTS Consulting used the formula:

$$(1 - \text{Cumulative probability of (beginning inventory)})$$

³ Two control sets of data were run through the Available Max with the (1- cum service level) opportunity calculation. On one input file, all fields but the cost field were kept constant, and in the other input file all fields but the demand field were kept constant. In both cases the model’s output was consistent. It ordered more of higher demand parts and more of low cost parts. In addition it ordered parts with the consistency and the magnitude which would be expected.

This statement would be the mathematical expression of the situation in Graph 1. The output from this equation provides the right side of the distribution (from 5 units and higher) while the following formula would provide the left side of the distribution (from 5 units and lower):

(Cumulative probability of (beginning inventory))

By minimizing the right side of the distribution, (1-Cumulative probability of (beginning inventory)), the first version of the model was using the correct concept for minimizing Expected Demand Not Satisfied not the Expected Additional Demand Satisfied. For execution purposes, the basic equation of (1-Cumulative probability of (n)) was altered so as to be more robust for the operational version. XLTS Consulting's improvement to the basic formula is called the Expected Additional Demand Satisfied purchasing equation.

EADS Purchasing Equation

Q = Incremental increase in inventory (order size)

n = beginning inventory

EXPECTED ADDITIONAL DEMAND SATISFIED = EADS

*EADS = -[(Q-1)*PROB.(n+1) + (Q-2)*PROB.(n+2)+...1PROB(N+Q-1)] + **Q[1-CUMPROB(n)]***

*Objective Function = Max (EADS/(Cost of Part * Q))⁴*

The EADS is a variation of the original formula, in that the right side of the equation (**in bold**) is identical to the original formula as presented to XLTS Consulting. The difference lies in the left side of the equation, which subtracts the current iteration (1,2,3,4, etc..) from the order size (Q) and multiplies this number by the current iteration added to the beginning inventory (n). This equation is performed for iterations from 1 to infinity until the outcome from equation is sufficiently close to 0. The model is set such that a computation of less than .000001 triggers the model to cease calculating this equation.⁵

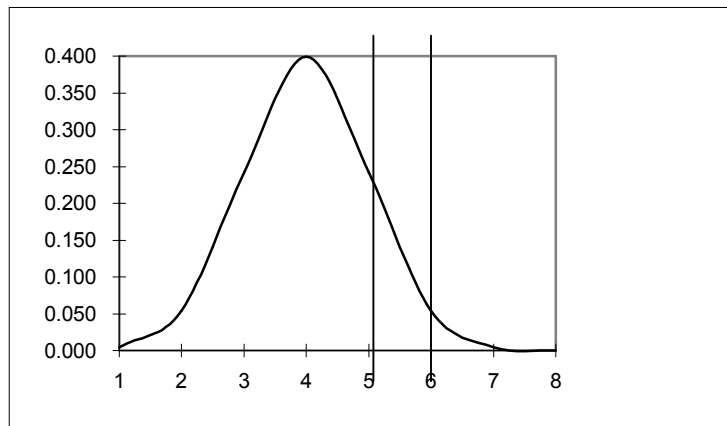
Rather than attempting to identify the uncovered or unprotected portion of the distribution curve (*the right side in Graph 1*), the EADS formula determines the size of the

⁴ This (EDS) formula and serves as the basis for two other derivations of the Avail Max decision system, one which handles returns and a second which estimates order fill.

⁵ .000001 was selected as it is reasonably close to 0. By setting this parameter, we save the model computation time which can be better utilized on relevant calculations. This parameter is especially important when the model is dealing with parts with higher demand patterns.

incremental increase in probability of demand (*the portion of probability between the lines from a demand of 5 units to a demand of 6 units in Graph 2*). This addition has benefits in terms of model operation, as well as increased accuracy of probability estimation.

Graph 2
Incremental Probability Added to Fill Rate by Moving from a Inventory of 5 to a Inventory of 6



Fill Rate Estimation

A second alteration to the Avail Max performed by XLTS Consulting was to the way in which the model estimates fill rate. As presented to XLTS Consulting, the Availability Max estimated fill rate by simply adding the probabilities of demand for the inventory which were in inventory. For instance, if the lead time demand was 4 units, and the inventory position of 3 units was chosen as an optimum holding amount, then the probabilities of demands of 1,2 and 3 units were added together to arrive at the estimated fill rate. If, for instance, the probabilities of demands for demands of 1, 2 and 3 were .20,.25, and .15 respectively, then the model would report a 60% fill rate for that particular part. XLTS Consulting changed the way the Availability Max estimated fill rate to a useful approximation of order fill by modifying the EADS formula explained in the previous section. This new fill estimation mimics how one would calculate fill rates on a spreadsheet. Table 2 provides an example of just such a spreadsheet fill rate estimation.

Table 2

Inventory	Demand	Probability	Demand Not Filled	Probability * Demand Not Filled	Probability * Demand
3	1	.1	0	0	.3
3	2	.25	0	0	.75
3	3	.35	0	0	1.05
3	4	.15	1	.15	.45
3	5	.1	2	.2	.3
3	6	.05	3	.15	.15
			Totals	.5	3

$$\begin{aligned} \% \text{ not filled} &= .5/3 = 0.16667 \\ \% \text{ filled} &= 1-.1667 = .8333 \end{aligned}$$

The Zero Demand Situation

A third area which XLTS Consulting improved the Availability Max was in the model's recognition of situations when there is no demand for a part. With any part, no matter how high or low the past period's demand, there is always the possibility that the part will experience zero. With the vast majority of parts in a spare parts database, this probability of zero demand is significant as most parts have demands of less than 2 units over a 2 week lead time. For example, a part with a Poisson distribution to its demand pattern which had a lead time demand the previous year of 2 units would have a 13.5% chance of not being subject to a demand the following year (*given use of the naive forecast*). Clearly, this would not amount to 13.5% fill rate estimation for that part. As the part experienced zero demand, any attempt at fill rate estimation is an illegitimate endeavor. As presented to XLTS Consulting the Availability Max added the probability of zero demand into its fill calculation. XLTS's improved Availability Max removes the probability of zero from the fill estimation.

The fill rate estimation is simply a modification of the EADS formula used to purchase parts. The same algorithm is used with 0 used as the beginning inventory variable (n) and the ending inventory substituted for the order amount (Q). The fill rate is then estimated by dividing the EADS by the mean demand of the past year.

Fill Rate Estimation

$$\begin{aligned} Q &= \text{ending inventory} \\ n &= 0 \end{aligned}$$

EXPECTED ADDITIONAL DEMAND SATISFIED = EADS

$$EADS = -[(Q-1)*PROB.(n+1) + (Q-2)*PROB.(n+2)+...1PROB(N+Q-1)] + Q[1-CUMPROB(n)]$$

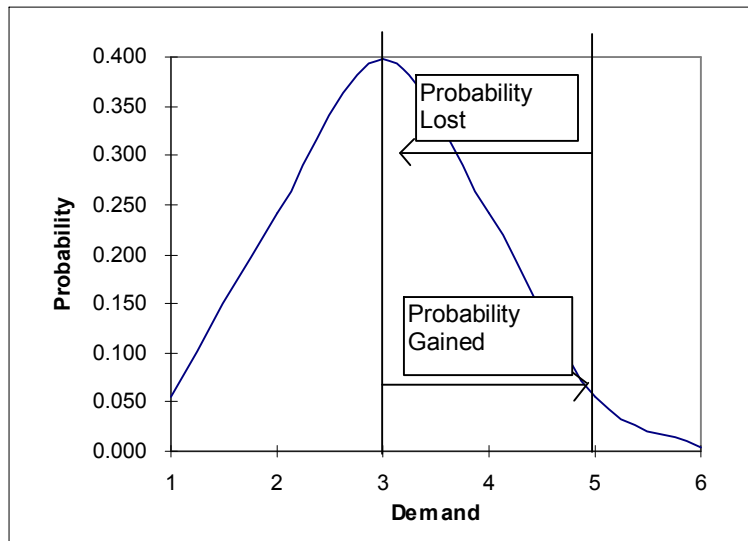
$$\text{Fill Rate} = EADS / \text{Mean Demand}^6$$

⁶ The Availability Max model contains both a global and individual fill rate cap which can be entered into the model's screens before the model is run. CASE wanted to achieving a global fill rate of 85%. This was entered before the model ran, and in addition, individual caps were set somewhat higher than that level. However, the minimums and package quantities in whose increments forced the model to purchase in larger increments meant that rarely were the individual fill levels close to the global or individual cap. It is important, when analyzing the

Returns

A third change made to the Availability Maximizer was to the method by which the model chose to return parts. When first presented to XLTS Consulting the model used the same logic that was in the original part purchase equation (EADS). However, this did not run in reverse (returns) very well. It displayed a tendency to minimize the right side (uncovered and unprotected) of the distribution as the current inventory was reduced by the order size. For the EADS modification which was applied by XLTS Consulting, (Q), this time the incremental decrease in inventory, is subtracted from the current inventory (n) to generate (z), the substitute factor for (n) to enter into the modified EADS equation.

Graph 3



From Graph 3, it is clear that the probability gained of moving from a demand of 3 units to a demand of 5 units (a purchase quantity of 2) and the probability lost of moving from a demand of 5 units to a demand of 3 units (a return quantity of 2) is identical. Therefore it is only necessary that the formula for a part purchase be modified to generate the probability lost of a return. This is generated by changing the semantics of (Q) in the equation from order amount to return amount. This is performed by subtracting the return amount from the current inventory (n)

model's output file, to remember that the caps do not limit the fill which a individual part can attain. They only prevent the model for purchasing addition pieces if the estimated fill rate is

and using the output from this activity (which we call (z)) to enter as a substitute for current inventory (n). This new output could then be called the Expected Demand Lost (EDL) as opposed to the (EADS).

The EADS Equation (EDL) Modified for Returns

Q = incremental decrease in inventory

n = current inventory

z = n - Q

EXPECTED DEMAND LOST = EDL

*EDL = -[(Q-1)*PROB.(z+1)+(Q-2)*PROB(z+2) +...1PROB(z+Q-1)] + Q[1-CUMPROB (z)]*

*Objective function = Min(EDL/(Part Cost * Q))*

Is This Logic The Correct Logic to Use for Service Parts Inventory Management?

Dr. Hau Lee, Professor of Industrial Engineering at Stanford University, viewed the Availability Max model in operation and recognized it as a application of the greedy heuristic.⁷ As it happens, Dr. Lee had jointly published a paper on the greedy heuristic's use in inventory management in which he supports its use for situations with large numbers of parts (a large number of parts in his opinion was over a thousand). In experimental results take from his paper Multi-Item Service Constrained (s,S)⁸ Policies for Spare Parts Logistics Systems published in Naval Research Logistics, Lee, Kleindorfer, Pyke, and Cohen used a multi-item algorithm with a Poisson distribution for both high and low demand types. 250 periods were simulated in order to reduce any random error. The results were that the greedy heuristic approximation was very accurate, with average errors ranging from .0006 to .031 for low service level requirements, and from .005, to .008 for high service level requirements. The following quote is from the Naval Research Logistics article.

*"It is possible to apply a greedy heuristic to both S (order up to level) and s (order point) incrementing with either S or s, for the part and control variable that provides the largest incremental increase in service for the minimum cost."
(570)*

above the cap on a particular iteration.

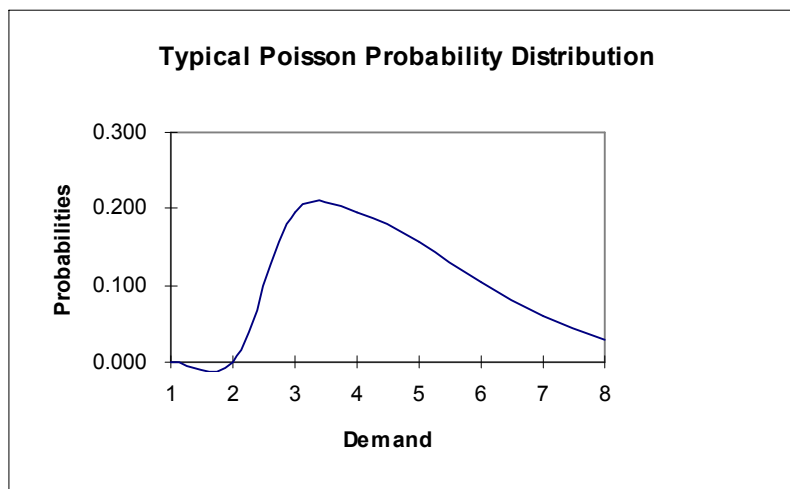
⁷ Lee, Pyke, Kleindorfer, and Cohen. "Multi-Item Service Constrained (s,S) Policies for Spare Parts Logistics Systems." Naval Research Logistics Vol. 39 pp. 561-577 (1992)

⁸ The notation s in (s,S) = reorder point, S = order up to point

The Poisson, the Normal and the Compound Poisson Distribution Assumptions and the Problem of Specification⁹

In order to develop the probabilities of different demands for different items for use in EADS, and EDL, it becomes necessary to choose a probability distribution which will closely fit the future expected demand. The Normal distribution is used when the demand is sufficient in volume such that the law of large numbers allows for accurate forecasting. (The graphical representations of the Normal distribution can be found in Graph 2 which is up a few pages.) However, for service parts only the smallest minority of parts fit this description. For the rest either a Poisson, Gamma, or Compound Poisson is conventionally believed to offer the correct approximation.¹⁰ The Poisson and Gamma are very similar leftward leaning, positively skewed probability distributions. The graphical representations for both are as follows in Graph 3.

Graph 3



¹¹The Poisson and Gamma are both positively skewed distributions (positively skewed means that the longer tail is in the positive number direction). They are typically used when there

⁹ The Problem of Specification defined as the attempt to fit a historical pattern to a probability distribution for the purposes of using statistical methods on the data. There are a few quantitative techniques such as the Lillefors test for normalcy, but more frequently the problem of specification is resolved through the application of probability distributions used for different situations taken from published works.

¹⁰ Another popular distribution is the Negative Binomial which is useful for approximating binary events. However, as the Compound Poisson is very similar to it, only the Compound Poisson will be analyzed in this paper.

¹¹ Continuous Distributions - specified outcomes can not be defined, but range of outcomes can be defined

is a high degree of randomness the historical data pattern. Both can be used to predict events like the timing of customers arriving at bank teller window, trucks arriving at a dock, in addition to the demand pattern for C items. The Poisson distribution has been extensively tested and found to be most effective at approximating future demand when the average lead time demand is below 10 units over the test period.^{12 13} The Compound Poisson distribution is used when the demand is both random and extremely “lumpy.” This distribution is especially applicable when items experience demand in conjunction with one another, for instance, the demand of a left shoe with a right shoe, or the demand for complementary repair parts. The problem with the Compound Poisson is in its calculation which is complex. In most low demand situations, either the Poisson or Compound Poisson can be used effectively, and it was the ease of computation which was the deciding factor in favor of the Poisson for the Availability Maximizer model.

When the model was first presented to XLTS Consulting, it only used the Poisson distribution. XLTS Consulting added in the use the Normal distribution for parts with more than a historical demand of 10 over the replenishment lead time. The Normal distribution is calculated in the Availability Maximizer through the use of polynomial exponents displayed below. Polynomial exponents are simply a method for approximating the Normal distribution given a certain normalized value for x.

Polynomial Exponential Approximation for the Normal Distribution

$$k = ((\text{beginning inventory} + Q) - \text{mean demand}) / (\text{standard deviation of demand})$$

for

$$(0 \leq k \leq \text{infinity})$$

$$1 - .5(1 + .196854 * k + .115194 * k^2 + .000344 * k^3 + .019527 * k^4)$$

Discrete Distributions - specified outcomes can be defined, and range of outcomes can be defined.

¹² Archibald, B., E. A. Silver, and R. Peterson (1974). “Selecting the Probability Distribution of Demand in a Replenishment Lead Time.” Working Paper No. 89, Department of Management Sciences, University of Waterloo.

¹³ The Availability Max model does not operate under any lead time parameters. It simply analyzes the demand it is fed as demand over some interval, the manipulation for the purposes of adjusting for lead time are performed on the input file. The project team is currently using a baseline of a two week total lead time (review + replenishment) which means that all parts with demand less than 234 per year fall into the Poisson assumption. This means that for a typical dealer, less than 100 parts will fall into the Normal calculations in the model.

Minimums and Package Quantities and Return Thresholds

The model's logic for choosing parts to buy and hold is known as the "greedy heuristic." However, while it is single minded in its search for the best opportunity, it may create purchasing scenarios that are uneconomical. For this reason, XLTS Consulting helped insert a minimum order quantity on the input file. The minimum order quantity was based on a EOQ with a order cost of \$5 and a holding cost of .24 per year.¹⁴ In addition, in order to guarantee orders consistent with the client's system, a package quantity column was entered into the input file.¹⁵ Both minimums and package quantities are used when deciding how much to buy. The first purchase will always be in the minimum order amount, and then successive purchases will be in increments of the package quantity. However, When returning parts, the minimum field is not used. To ensure that the model did not return parts that may be needed at another time, a third column was added to the input file.¹⁶ This column was generated as a 9 month supply of yearly demand. This was called the return threshold field.^{17 18}

¹⁴ Variable order costs (r) and holding costs (A) are recommended as it is generally difficult to pinpoint actual costs. For this reason Silver and Peterson recommend creating exchange curves displaying the effect on order frequency and cycle stock \$ with various A/r fractions. At CASE, while 24% holding cost is un-controversial, the order cost is subject to discussion.

¹⁵ In order for the model to operate correctly, minimums are always entered as a multiple of package quantities.

¹⁶ During the project, CASE voiced a need for the model to deal with non-quantitative issues, or issues which were not feasible to put into a mathematical form. These included substitutions, multi-substitutions and unit of measure issues. The substitution issue dealt with the transfer of demand data from a old part which had been in some way improved and thus been given a new part number. In some cases one part may be reengineered into two parts or two parts reengineered and combined into one part. These are defined as multi-substitutions. As for unit of measure issues, it was common for the dealer and XLTS's client to have incompatible data records. For instance if a hose is regularly sold in 50 foot lengths, the demand data may be corrupted when a sale of one 50 foot length is reported as a sale of "50" which may be interpreted as a sale of fifty 50 foot lengths. These types of issues were left to "post processing" in which the data from the output file would be analyzed on an exception basis

¹⁷ One outcome from all of these changes is that the model was altered to better fit the clients' day to day needs. A second outcome is that the degree of optimization was effectively reduced as more constraints were placed on the final outcome of inventory purchases and returns. Between individual parts the fill rates became more staggered, there were many parts with 99% fill rates reported, and fewer parts with midrange fill estimations of 83, 86, 92% etc... With these added constraints the model chose to leave many parts with no fill rate and others with fill rates well beyond the 85% target.

¹⁸ The model has no time horizon or time orientation. It accepts whatever demand it reads from the input file as the demand over the interval it is calculating. If demand over 5 years were on the input file, then the model would calculate a optimal purchase quantity for a five year period. As we have assumed a two week total lead time (1 week for review and 1 week for replenishment), the yearly demand was divided by 26 in order to arrive at the demand over lead time. In addition, the standard deviation, which is used in computation for the probability of demand the higher demand parts, was available to us from the clients information systems on a monthly basis. In order to

Forecasting

The focus of the project on which the Availability Max model was developed was to test the inventory replenishment logics for the purposes of selecting a professional software package which would perform functions similar to the Availability Max. It was decided by the team members that the model would be fed as naive forecast, and that when the software for inventory replenishment was selected, a software package for forecasting would be selected. This basic naive approach was further augmented in order to capture the seasonal nature of the parts of the client. The naive approach was augmented as the following paragraphs explain.

a. For parts with average annual dollar volume $x \leq \$10$

If part has demand of 6 months, then looking forward 1 and 2 years ago use the total of 12 months of demand divided by 2 in order to generate the bi-monthly demand forecast.

b. For parts with average annual dollar volume $\$10 < x < \300

If part has demand of 3 months looking forward for 1 and 2 years ago use the total of 6 months of demand divided by 2 to generate a bi-monthly demand forecast.

c. For parts with average annual dollar volume $> \$300$ parts

If part has demand of 2 months looking forward for 1 and 2 years ago, then use the total of 4 months of demand divided by 2 to generate a bi-monthly demand forecast.

When the forecasting software is finally chosen, this methodology would no longer be used. However, the spare parts databases promises challenges which must be dealt with. The vast majority of parts would be classified as C items under traditional inventory theory, and according to Silver and Peterson, C parts do not lend themselves to anything but naive forecasts. However, for a small segment of the database, there are parts which can be forecasted reasonably well. When we say “reasonably” we mean better than a 25% forecast error.

Conclusions

In testing, the Availability Max purchased both inexpensive parts and higher demand parts. Spreadsheets which mimicked the logic in the Availability Max were used to test the ordering and return amounts as well as the corresponding fill rate calculations. These tests to the model’s operating logic indicated the model was selecting parts in conformance with its programming. As of the time of this writing the largest issue is the size of the order minimums.

change the standard deviation to a bi-monthly variance the monthly variance was divided by the square root of 2.

After preliminary runs, the model appeared to be ordering up to the minimum level for the majority of parts. There is evidence that these minimums may be set too high, even though the order cost used is only \$5 per line.

During the development of the Availability Max, it was a common occurrence for extra requirements to be projected upon the model. It was XLTS's position that while it may often be intuitively appealing to attempt to include all inventory considerations into the model through the addition of parameters, there are two drawbacks to this approach. Number one, the attempted optimization of more than a few basic parameters can lead to a "middling effect" whereby the parameters tend to neutralize one another. Number two, with each additional parameter a level of complexity is added to the modeling process. This is undesirable as it requires additional resources from the development team. Second, in developing a day-to-day operational inventory management system, simplicity of execution is a necessity.

A DESCRIPTION OF THE SECOND TYPE OF SPARE PARTS INVENTORY
METHODOLOGY: *SPARES OPTIMIZER*

System Context

The Second inventory tool to be analyzed is the SPARES Optimizer which was developed by XLTS Consulting in conjunction with professors from the University of Pennsylvania, Stanford, IBM's T. J. Watson Research Center, and IBM's National Service Division for dealing with the complex environment of IBM's service parts needs.¹⁹ SPARES can support both planning and operational functions. SPARES is particularly interesting as it allows for pooling, which is the sharing of inventory between geographically separate inventory sites, across any echelon level.²⁰ Pooling can allow more than one inventory location to act as a single facility.

A second difference in SPARES is its ability handle multi-echelon environments. Although most all inventory situations are multi-echelon, it is more common for inventory to be managed at each level, with single echelon inventory optimization being the goal. The decision system that can incorporate all echelons is rare. According to its developers SPARES is "...a logistics tool that optimizes spare parts stocking policy to meet the client's product service levels while minimizing cost." SPARES was developed in response to two basic business needs:

- 1. For managing flexibility-in setting strategically driven service targets for different market segments*
- 2. Improved inventory efficiency and cost control²¹*

SPARES was developed to deal with a very extensive distribution network. The IBM network which SPARES was first modeled on had a distribution network as follows.

1. 2 Central Warehouses

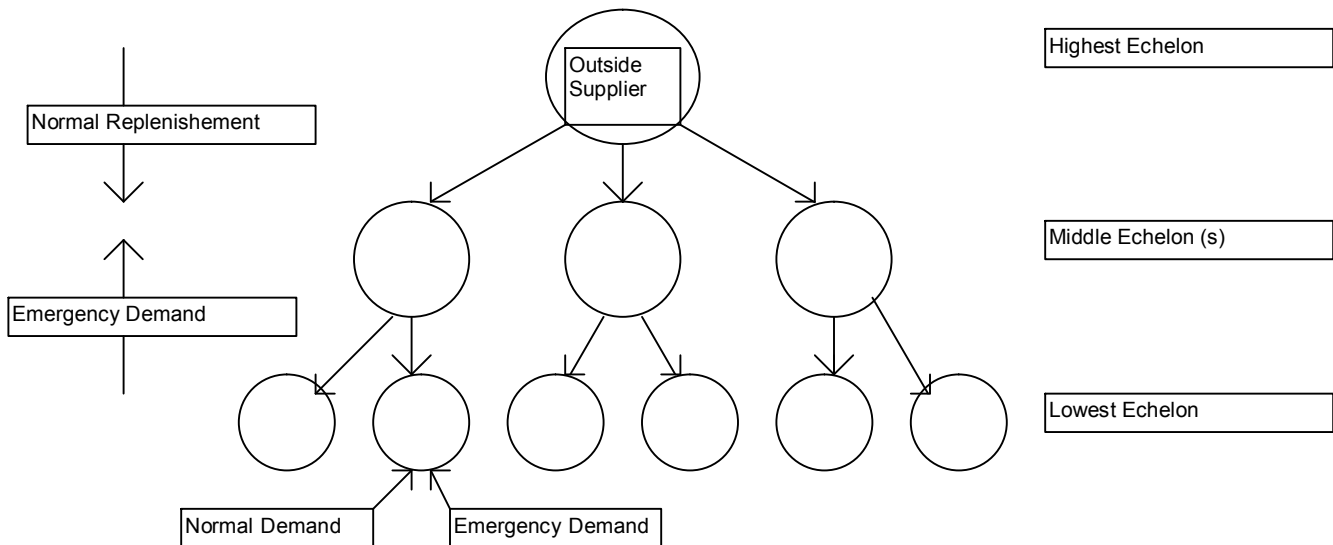
¹⁹ A literature review was performed at the time of SPARES development. According to the developers, "...most of the relevant existing multi-echelon theory was based on one-for-one replacement logic which was the basis of military logistics control. While appropriate for low demand items, the policy did not provide adequate cost/service performance for the wide range of demand rates which were present in the client's environment. In addition, these models treat items independently and hence the product/part service interactions are not captured."

²⁰ According to the developers of SPARES, "...pooling has a significant impact on stocking policy at the echelons above pooled echelons. Pooling increases the amount of emergency demand that is satisfied at a node, thereby shifting some of the passed up emergency demand to backorder instead. The net result of pooling at an echelon is lower stocking levels at the next higher echelon.

²¹ Cohen, Kamesam, Keindorfer, Lee, Tekerian. "OPTIMIZER: A Multi-Echelon Inventory System for Service Logistics Management." IBM Research Division Publication: 1991.

2. 21 Field Distribution Centers (Regional DCs)
3. 64 Parts Stations
4. 15,000 Outside Locations

Inventory Network Under Which SPARES Was Developed
(Customer Demand May be Normal or Emergency, Emergency Demand is Expedited in Stockout Conditions) Figure 1.



In addition, there were over 200,000 part numbers in the IBM spare parts inventory database. In the IBM network, part numbers were tracked at every stocking location. Because of this SPARES was developed to create service objectives at each echelon level, it also has a coding for critical parts called TCGs. The failure of a TCG part leads to the complete failure of the machine.²² Cost functions used by SPARES included the following

1. *Replenishment Costs*
2. *Emergency (Expedite) Costs*
3. *Inventory Holding Costs*

The model included three separate scenarios

1. *One part - one location model*
2. *Multi-product - one location model*
3. *Multi-product - multi-echelon model*

²² Criticality was a consideration for the Availability Maximizer client. The plan was to create a binary field on the input which would be read by the model as a “must carry” part regardless of whether the cost drivers or demand history supported the purchase of the part. The problem the team encountered was the client had not coding for critical parts in its system.

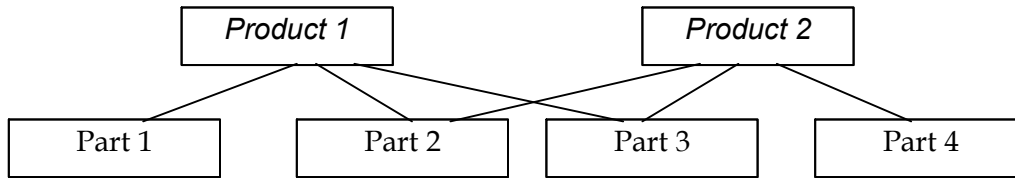
SPARES takes a network approach where each inventorying location is represented as a node. Each of the nodes is serviced by a higher echelon node from which it places and receives “normal” replenishment orders and to which it expedites “emergency” demands in stockout occasions. This highest echelon, usually, a major warehouse node is serviced by an outside supplier (Figure 1). Nodes in SPARES can be of three types. A regular node represents a normal stocking location. An aggregate node representing a group of locations within an echelon. Aggregates are useful when lower echelons have hundreds or thousands of locations. Finally, a dummy node, which stocks no parts is an artificial node set up to pass demand directly from the echelon below the dummy to the echelon above. Dummy nodes are useful in modeling locations in middle echelons that satisfy their local customer’s demand.

Normal customer demand to a bottom echelon node and replenishment orders to a middle echelon node are satisfied from stock when possible and backordered in the case of stockout. Emergency customer demand is handled differently. Emergency demand is also satisfied from stock when possible, but is not backordered in stockout situations. Instead the order is fulfilled first from a node’s pool. If the node’s pool cannot fulfill the order, the demand is passed up to the node’s supplier in the next higher echelon. The expedited order is no longer considered part of the lower echelon node’s demand.

Product/Part Structure

Customer demand at the lowest echelon is represented as product order arrivals. In a field service setting these order arrivals are actually product failures. An order could result in the need for any of the component parts of the product. SPARES uses a product-to-parts structure as depicted in Figure 2.

Figure 2: Product Part Structure Example



The figure above displays the concept of parts being used in multiple products. SPARES may also be used to handle part related orders by setting up a one-to-one correspondence between products and parts where each product is composed of just one part. SPARES uses a part group concept where a single part represents a group of parts with similar characteristics, (e.g., value, usage, criticality, lead time etc..). This grouping of parts eliminates the need to model every part (an impossible task when parts number in the thousands).

Similar to the Availability Max, product order arrivals during review period are assumed to be random and to adhere to a Poisson distribution. However a more complicated assumption is made, that the number of particular parts needed is also random but based upon a geometric probability distribution.

Network Costs

Network costs for SPARES = ordering costs, holding costs, expediting costs, backorder costs, and pooling costs. Some of these costs have transportation costs which are separated out in the model. SPARES recognizes only these costs for use in the optimization algorithm. The first step in the algorithm determines inventory replenishment policy parameters for each part in order to meet its minimum fill rate. This fill rate specified by the user, is the percent of part's demand to be satisfied directly off the shelf. The algorithm then determines whether to increase part fill rates to minimize total costs. That is, it may raise stocking levels to reduce emergency expediting costs. Next the algorithm further adjusts each part's replenishment policy parameters to ensure that each product's service level is achieved. Cost minimization is maintained by increasing the part which is most cost effective in improving the product's service level.

Operation of the Model

SPARES requires 15 input files which can be grouped into 5 groupings:

1. *Physical Network Description*
 - a. *echelons*
 - b. *nodes*
 - c. *suppliers*
2. *Pool Group Data*
 - a. *part groups*
 - b. *minimum fill rates*
 - c. *part costs*
 - d. *external suppliers*
 - e. *transportation time*
3. *Product Data*
 - a. *product*
 - b. *order rate*
 - c. *service policy*
 - d. *product/part groups*
4. *Pooling Data*
 - a. *pool members*
 - b. *pool costs*
5. *Operating Parameters*

SPARES begins by computing a near optimal value for all parts at all locations in the first echelon. The algorithm then proceeds to the second echelon and repeats the calculation. At the second echelon locations, however, the incoming demand distribution is no longer determined from the base forecast. Instead the distribution is determined by selections made in the first echelon. There are three types of incoming demands which can be experienced at each location of the second echelon.

1. *Replenishment demand from locations at the lower echelon that are sourced by the second echelon location.*
2. *Emergency demands generated by part failures from locations in the lower echelon which are supported by second echelon locations*
3. *Failures induced demands from customer machines that are supported directly by the location*

Incoming demand distributions to a particular location at the second echelon are then obtained by aggregating the first two moments from all the locations at a lower echelon that are supported by this current location, as well as those from the direct customer demands at the location.

Ordering decisions are made based upon the results of the periodic review (weekly, daily, etc.) of inventory. If a part has fallen below its replenishment policy reorder point (Min) an order is placed to raise the inventory to the reorder limit (Max). Each part stocked at a location has a order point and reorder limit which relates to its safety stock, cycle stock, and fill rate.

Conclusions

SPARES produces an output file with stocking policies, fill rates, and cost components for each part and its associated part group. One issue which surfaced very rapidly, and has analogies to the Availability Max model was that when SPARES was run, the value of the total inventory purchased by the system was much smaller than initially expected. Upon closer inspection it became apparent that the model was meeting its service objectives by purchasing inexpensive, non-functional parts in order to achieve the desired service level.²³ Certainly, using the greedy heuristic will provide a solution which requires augmentation whenever the lower cost parts in the database are of lower priority. In order to overcome this, priority or critical coding was required to be entered on the input file. The instructed the computer to execute the greedy heuristic subject to the constraint of carrying a certain number of critical parts.

A second issue with SPARES was the instability from week to week in stocking levels. While mathematically the optimal carry amount may be to vary the stocking levels in time with demand, however in reality, a more stable stocking level is necessary to mitigate variable conditions.

The final SPARES system consisted of four modules

- 1. Forecasting System: A batch of programs which estimated failure rates of individual part numbers in each product. These failure rates were then used with machine population to come up with the expected need.*
- 2. Data Delivery System: Processes over 15 gigabytes of data to provide the basic data inputs for SPARES*
- 3. Decision System: solves the multi-echelon stock control problem. The module has dynamic memory management scheme to control the virtual memory allocation, release and garbage collection of storage for the data structure.*
- 4. PIMS Interface System: serves as interface for the output of the Decision System and the PIMS*

Outcome of SPARES

- 1. Reduction in required inventory (in the test case 20 to 25% less inventory)*
- 2. Improved service*
- 3. Increased responsiveness of the system*
- 4. Enhanced flexibility in responding to changing service requirements.*

²³ This will always be a problem with the greedy heuristic when lower cost parts are less critical, in general, than higher cost parts. With the Availability Max client, there was no coding for critical parts, although it was a consistent interest of the client to incorporate criticality into the ordering policy.

Benefits Linked to

1. *Optimization*
2. *Improved Forecasting*
3. *Multi-Echelon Linkages*
4. *Product-Part Interactions*

One thing to remember with SPARES is that there is an adjustment period before the full benefits are realized.

Appendix A

XLTS Consulting made improvements in three logic areas of the Availability Maximizer. These were all made through the application of a single algorithm which can be used for either buying, returning or calculating the fill rate of service parts. The three different uses of the formula are explained below.

The Expected Additional Demand Satisfied algorithm can be used for part selection, fill rate estimation, and returns

When used for part selection

Q = Incremental increase in inventory

n = beginning inventory

EXPECTED ADDITIONAL DEMAND SATISFIED = EADS

$EADS = -[(Q-1)*PROB.(n+1) + (Q-2)*PROB.(n+2)+...1PROB(N+Q-1)] + Q[1-CUMPROB(n)]$

Objective Function = Max (EADS/(Cost of Part * Q))

When used for fill rate estimation

Q = current inventory

n = 0

EXPECTED ADDITIONAL DEMAND SATISFIED = EADS

$EADS = -[(Q-1)*PROB.(n+1) + (Q-2)*PROB.(n+2)+...1PROB(N+Q-1)] + Q[1-CUMPROB(n)]$

Fill Rate = EADS / Mean Demand

When used for returns

Q = incremental decrease in inventory

n = current inventory

$$z = n - Q$$

EXPECTED DEMAND LOST = EDL

$$EDL = -[(Q-1)*PROB.(z+1)+(Q-2)*PROB(z+2) + \dots + 1*PROB(z+Q-1)] + Q[1-CUMPROB (z)]$$

$$\text{Objective function} = \text{Min}(EDL / (\text{Part Cost} * Q))$$

Appendix B

GLOSSARY of Terms from the Input File

On Hand QTY - Current inventory which is at the dealer location

ON_ORD_QTY - Parts on order, Availability Max simply adds this to current inventory

sales - A two week interval of demand as defined in issue 6 in this document

DLR_NET -Cost of part to the dealer

RTN_CD - A proprietary code used by GPN to designate the return status of a part. This is currently not used in the Availability Max although there is a column for this input. There are eight different codes

Space - returnable

1 Not Returnable - non returnable short shelf life

2 Not Returnable - obsolete

3 Permanent returnable - always returnable

4 Non-Returnable current yr+1 - non returnable in the future 1 years

5 Non-Returnable current yr+2 - non returnable in the future 2 years

7 Non-Returnable - ship direct only

%std_dev_sal - The monthly variance of the part divided by the square root of 2 to provide a bi-monthly figure

Min_order - The EOQ based upon the cost drivers specified in issue 7 and 19 in this document

Package_qty - When used as on the purchase size the package quantity is the purchase increment above the minimum. When used on the return side, the package quantity is the return increment down to the return threshold.

Target_Serv_level - This is the maximum service level which a individual part may attain - this is only used for the purposes of purchases

Return_threshold - This is the minimum piece amount a part may be returned to. For instance a part with a current inventory of 10 and a Return_threshold of 3 may return 7 pieces of this part

Appendix C

Technical Information

SPARES Optimizer is written in “C” programming language and runs under DOS Version 3.0 or later. It can be run on any IBM with a minimum of 1MB of RAM

Inputs and outputs are ASCII files

The maximum model parameters are

1. Echelons: 6
2. Nodes: 150
3. Number of Part Groups: 100
4. Number of Pools: 50
- 5: Number of Products: 40