APPLICATION OF SIMULATION AND ZERO-ONE PROGRAMMING FOR ANALYSIS OF NUMERICALLY CONTROLLED MACHINING OPERATIONS IN THE AEROSPACE INDUSTRY

William R. Lilegdon II, Charles H. Kimpel
Pritsker & Associates, Inc., P.O. Box 2413
West Lafayette, IN 47906 USA

Donald H. Turner
Booz, Allen & Hamilton, Inc.
8800 Pleasant Valley Road
Cleveland, OH 44131 USA

ABSTRACT

This paper presents a simulation analysis performed under Project 2108 "Integrated Sheet Metal Fabrication System" of the United States Air Forces' Integrated Computer Aided Manufacturing (ICAM) Program. The overall objective of the ICAM program is to increase productivity in the aerospace manufacturing community through use of advanced technologies and decision support systems. The objective of Project 2108 is to design a state-of-the-art Integrated Sheet Metal Fabrication System, build it on site at a major aerospace location and track savings. The purpose of the simulation study described herein, is to evaluate various design alternatives for an aluminum sheet metal fabrication system. In this application a zero-one algorithm is called by the simulation each time a decision is needed for release of orders to the manufacturing cell. The paper presents the system description, modeling approach, and results of the analysis effort.

INTRODUCTION

A numerically controlled sheet metal router and its supporting systems were the focus of this analysis effort. A simulation model of the system was developed using SLAM II™. The purpose of the study was to examine the alternative technologies for their impact upon the throughput and unit production cost of the manufacturing cell. Two types of technologies were investigated. The first alternative was increasing the sophistication of the sheet metal router so that it can route variable sized stacks (length, width and depth) of sheet metal rather than stacks of fixed size. The second type of technology analyzed was utilizing computer aided selection of orders to increase the efficiency of "nesting" sheet metal parts on the sheet metal stacks. The result of more efficient nesting is reduced material scrappage and reduced inventory carrying costs. Zero-one programming was used to model the computer aided selection of orders and was called directly from the SLAM II simulation of the numerically controlled routing machine cell.

SYSTEM DESCRIPTION

The product of the sheet metal router is flat sheet metal parts of irregular shapes as shown in Figure 1. Parts are machined on the router in stacks as shown in Figure 2. A typical stack is

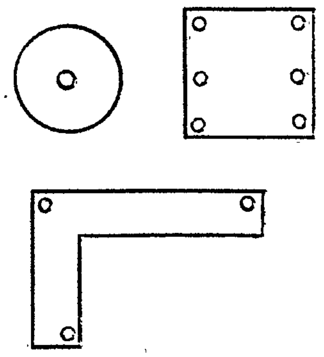


Figure 1. Sample Trumpf Parts.

---

*SLAM II and SDL are registered trademarks of Pritsker & Associates, Inc., P.O. Box 2413, West Lafayette, IN 47906 USA.

Proceedings of the 1982
Winter Simulation Conference
Highland * Chao * Madrigal, Editors

82CH1844-0/82/0000-0281 $00.75 © 1982 IEEE

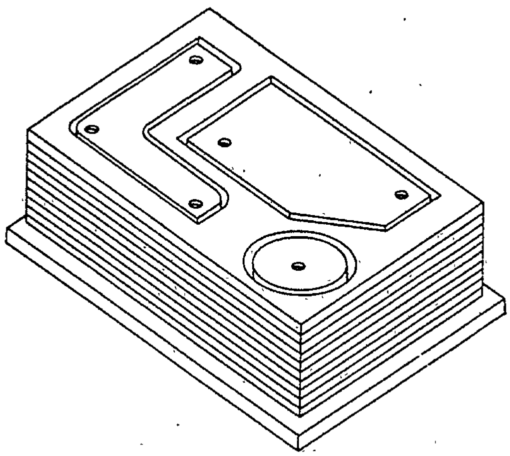


Figure 2. Aluminum Stack as it Leaves the Router.

4' wide by 6' long by 1/2" high. The number of sheets of metal in the stack depends upon the gage of the material. For .050" gage material, ten sheets of material are required to form a .5" stack. Depending on the size of the parts to be routed, 1 to 75 parts may be nested on a stack (Figure 2 shows a stack with just three parts). All the sheet metal in a stack must be the same material, temper and gage. Hence, all parts produced from routing a stack of material are from the same part family.

Order release initiates the process of routing a sheet metal part. Orders are released at various times for several different manufacturing programs and, within each program, orders are released at various times for different subassemblies. Each order is released to the shop well ahead of its due date. When the order reaches the shop it is placed in a file with other orders of the same part family. The order that is closest to its due date determines the material, temper and gage of the next stack to be produced. This order is called the critical order.

The stack is formed by nesting the critical order and other parts in the part family file of the critical order, until the sheet metal stack is utilized as fully as possible. In the nesting activity, operators arrange computer images of the parts on a computer screen. Through manipulation of the images, the operator attempts to maximize the number of parts on the nest and, achieve a satisfactory material utilization. Once completed, the computer image of the nest is converted to NC instructions by the nesting computer. These instructions will reside in memory until the Trumpf operator requests their transfer to the router controller.

Machining of ordered parts begins with retrieval of the material shearing the material to size and stacking the aluminum sheets on a baseplate (typically a .250" thick aluminum sheet that is reusable). The sheets and baseplate are then mechanically clamped together and loaded onto the router table. The purpose of the mechanical clamp is to hold the sheets to the baseplate until sufficient rivets have been placed by the router to hold the stack together. After loading the stack, the operator calls the nesting operator and directs the NC instructions for that nest to be "downloaded" into the router controller. The machine cycle is begun, performing riveting, routing, and drilling. When the cycle is complete the stack is unloaded from the Trumpf, the completed parts are part separated from the lattice (the scrap material), wiped dry of cutting fluid, and taped together for transportation to subsequent degreasing and deburring operations. A single operator is required to clamp the material, download the numeric codes, change drill bits, monitor machine operation and remove parts after all machine operations are completed. Figure 3 summarizes the activities required for production of the sheet metal parts. Figure 4 shows the numerically controlled router configuration.

SIMULATION TECHNIQUE

The Trumpf system described was modeled using the SLAM II methodology. SLAM II was chosen because of its network-discrete event capability and its interfaces with graphics and database software. The modeling approach adopted was a combined network-discrete event. The production floor portion of the model was developed as a network taking advantage of speed and ease of modification of network modeling. The other portions of the model were developed as discrete events. The discrete event subroutines allowed easy substitution of different logic in areas of order releasing and nesting. A modular approach was adopted to represent the three phase activity of the system.

Production control activities were simulated using discrete events. Generic routines were developed to create entities representing orders from all air frames and assign them necessary physical parameters. To completely describe an order, part number, material type, temper, gage, due date, perimeter, area, rivets, interior drill holes, production quantity required, and air frame number were assigned.

Assignment of these attributes was made using the SDL-SLAM II™ interface. The attribute values were obtained from a major air frame manufacturer and maintained in a SDL relation database.

The nesting of orders for release to the production floor was accomplished using a network-

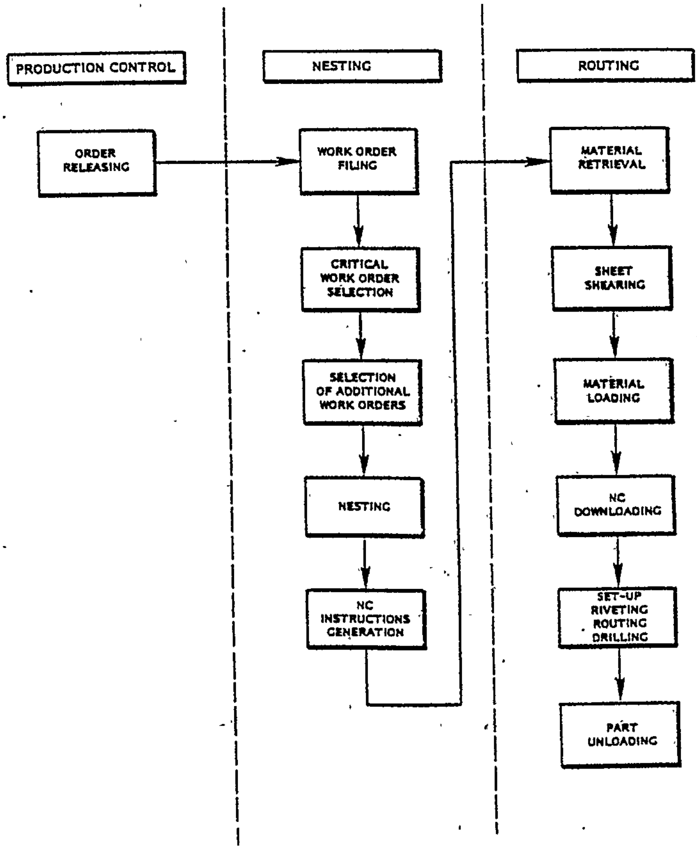


Figure 3. Trumpf Activity Breakdown.

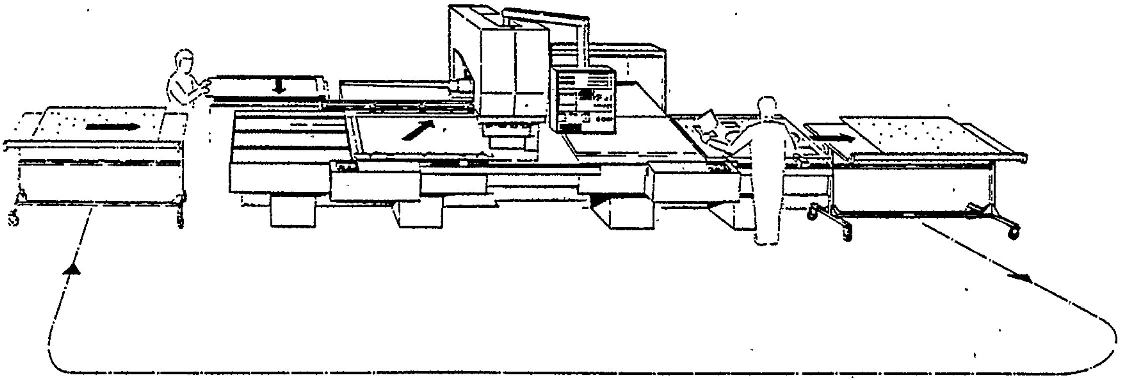


Figure 4. Trumpf Routing Cell.

discrete event approach to allow complex nesting logic. Initiation of the nesting activity occurred every eight simulated hours. This represented the arrival of the nesting operator, for the new shift. At this point the operator would poll the system to evaluate Trumpf loading. If five orders were in some stage of processing at that time on the production floor, no nest was prepared. This system control was established to prevent overloading of production network and to allow selection of orders from part family files to be performed within a reasonable range of the production time.

If loading of the production floor was below the five order limit the nesting event was initiated. The three phases of nesting work orders were simulated using three subroutines called through the SLAM II event code structure. The first phase of nesting, determination of critical part family file, was achieved by simply polling the first member of each part family file. A part family, as previously described, consists of a unique combination of material type, temper and gage. For the database of orders provided, 105 part families were developed. The part families were maintained as ranked queues by SLAM. With the identification of each entity, the entity was filed into the appropriate queue. Ordering of entities (orders) in the queue was performed by due date. The hottest or due date closest to the current time was at the front of the queue. With identification of the critical part family file order selection could begin.

In the baseline analysis a top-down search of the part family file was performed just as in the real system. The area required by the current order was compared to the area remaining to be nested. If a fit was possible the rank of included order was noted and the available area adjusted. The search continued, sequentially, until every member of the part family file had been examined.

After selection of the orders, the third phase of the nesting event began. Creation of a new type of entity, a nesting packet, was performed. In the baseline analysis the orders selected were removed from the appropriate file and the nesting packet attributes describing the production times and physical parameters of the nest calculated. Nest packet entities were described by nest number, material type, retrieve stack time, shear and delivered time, down load time, load time, set-up time, rivet time, route time, drill time, and clean-up time attributes. Each of these attributes were assigned and the entity was entered into the production floor network.

The multiple stack type analyses required determination of stack type to nest, before removal of the orders. The comparison of stack types was completed using a lowest cost per part criterion. Each set of orders selected was used to calculate the cost of producing that nest type. Cost per part was compared for all alternatives and the low cost stack type was selected. Appropriate orders were removed from the files and the nesting packet entity was created as specified.

The production floor system was modeled using a SLAM II network. Resources were defined as the operators, the hand clamp, and the Trumpf. Use of network allowed faster creation and communication of the Trumpf cell system.

MULTIPLE SHEET ANALYSIS

The first analysis performed with the Trumpf system model was a comparison of single stack type to multiple stack type routing. The baseline case examined the performance of the Trumpf if only sheets of 4' x 6' were stacked .50" high. With hardware modifications of the Trumpf router, multiple sheet sizes and stack heights could be used. Table 1 shows the sheet sizes and stack heights considered.

Table 1. Stack Types Nested.

Sheet Size	4'x2'	4'x4'	4'x6'	4'x8'
Stack Height	.250"	.375"	.50"	

The multiple stack option analysis showed significant performance improvement over the baseline (fixed stack size) analysis. The results are displayed in Table 2. Cost savings at 33 cents per part for 300,000 parts per year will, in a year, generate cost savings of $99,000 (a savings of greater than 10 percent in labor, material and inventory costs). The improvement is due to utilization of the smaller 4x4 and 4x2 sheet sizes and smaller stack heights to reduce material scrappage for part families with few orders.

The smaller stack heights reduced the number of excess parts produced, thus cutting inventory costs. Larger stack sizes permitted higher throughput and reduced set-up costs for part families which have many orders to be produced. The utilization of the various sheet size and stack height options are displayed in Figures 5 and 6.

Table 2. Multiple Stack Type Analysis and Cost Breakdown.

	Analysis		
	Cost $/Part	Throughput Part/Hr.	Average Mat. Util.
Single Stack Type	3.03	64.45	.5881
Multiple Stack Type	2.70	61.00	.6452

	Cost Breakdown			
	Material $/Part	Labor $/Part	Inventory $/Part	Total $/Part
Single Stack Type	2.60	0.29	0.15	3.03
Multiple Stack Type	2.27	0.30	0.13	2.70

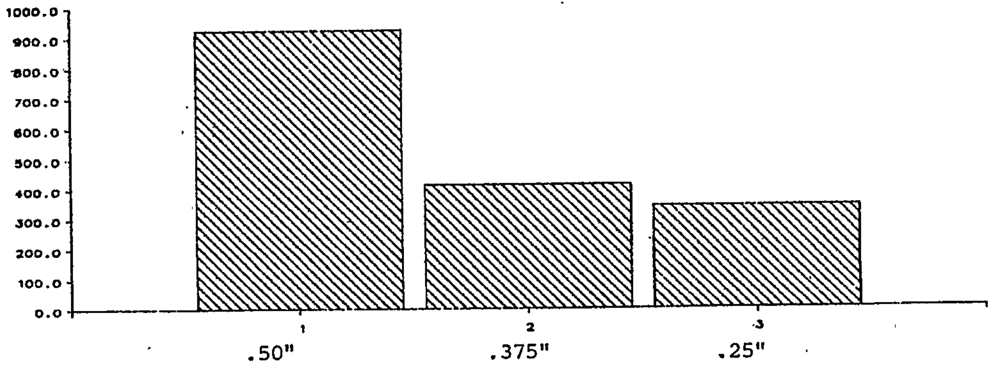


Figure 5. Multiple Stack Type Analysis-Stack Height Nested.

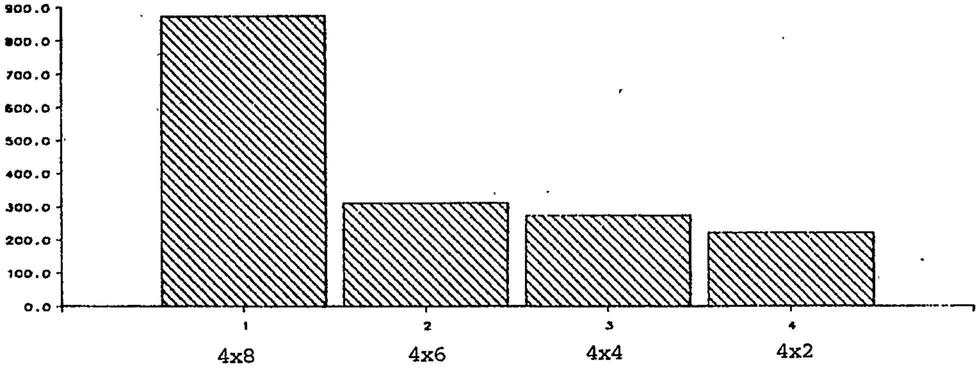


Figure 6. Multiple Stack Type Analysis-Sheet Size Nested.

ZERO-ONE FORMULATION

The current method of order selection in the nesting of orders for the Trumpf is the linear search. The definition of the selection task, however, is applicable to linear programming. The second phase of the Trumpf system analysis was to examine potential benefits of an optimized order selection procedure. The benefits of the optimized order selection procedure are to be measured in terms of improvement of performance parameters of cost per part and throughput.

Formulation of the order selection activity as a linear program required examination of the drivers of the system performance parameters. The cost of a part produced in the baseline case with linear search selection techniques was primarily material. Figure 7 shows the percentage breakdown of cost per part of the baseline case. With over 75 percent of the part cost being in material, more efficient material utilization should reduce cost per part.

Throughput of the Trumpf cell is a function of the number of nests produced and the quantity of parts on a nest. The more parts loaded onto a stack, the higher the system throughput. Therefore, material utilization is also a driver of throughput. With this assumption the formulation (Figure 8) was developed to maximize material utilization for a given stack type.

ZERO-ONE FORMULATION IMPLEMENTATION

The formulation of the order selection procedure attempts to maximize the material on a nest. Implementation of the formulation required development of an algorithm to solve the specific problem. The algorithm is executed in the simulation model each time a nest is created. Therefore, considerations of execution time and storage space, connoted implicit enumeration as a solution technique.

Reformatting the formulation for use with an implicit enumeration technique revealed that in a refined form (Figure 9), the formulation was unconstrained optimization. Maintaining the assumption that improved material utilization would improve system performance, the reformatted formulation was adopted to be solved by explicit enumeration.

An explicit enumeration routine was developed using nested do-loops. The consideration of execution time became a factor in the size of the part family file evaluated. The size of the part family files during execution had a maximum of over 400 members. Explict enumeration of a file of this size would require examination of $2^{400}-1$ possibilities. To reduce the number of combinations, the orders would be considered in groups of 10. In other words the best selection would be made with the first 10 orders and the remaining available space determined. The available space would then be optimally allocated to the next group of 10 orders. This process was repeated until all members of the part family were examined.

OPTIMIZED ORDER SELECTION ANALYSIS

The zero-one order selection analysis was simulated for both the baseline (fixed stack) and variable stack router configurations. The results are displayed in Table 3.

Zero-one order selection for the fixed stack router configuration produced significant savings over the current linear search procedures. More effective selection reduced the number of nests required to make the same quantity of parts. Throughput was increased and inventory was decreased.

Material costs were reduced by 4 percent. Figure 10 compared the distribution of material utilization per nest for linear and zero-one selection procedures. From the figure, the average material utilization decreases slightly for optimized selection but the average number of orders per nest increases.

Zero-one order selection did not improve performance of the variable stack router configuration. Material utilization was improved, but the higher utilization was due in part to selection of smaller sheet sizes in the zero-one analysis than in the current linear selection analysis. The smaller sheet sizes used in the zero-one order selection analysis offset savings in material utilization with increased set-up costs.

A more sophisticated zero-one formulation is required to aid order selection for the variable stack router configuration. As a minimum, set-up costs need to be added to the objective function in addition to material utilization. However, the zero-one formation does show some potential as it reduced material costs by 4 percent in the baseline (fixed stack) analysis.

CONCLUSION

Two types of improvements to the numerically controlled router were simulated. Upgrading the router to handle variable stack sizes produced cost savings of over 10 percent and is highly recommended. Upgrading the order selection is not recommended at this time. Further work is required to produce a sophisticated zero-one formulation that can produce significant cost savings.

Interfacing the zero-one programming algorithm with the SLAM II simulation of the routing system was implemented easily. This project demonstrated the potential for interfacing zero-one algorithms as well as other types of decision support software (e.g., Materials Requirements Planning [MRP] or Manufacturing Control Material Management [MCMM]) with simulation. A manufacturing system where decisions or schedules are provided by support software can be simulated by having the simulation call those support software packages (with proper input parameters) during the execution of the simulation. In this manner, a system which uses

CELL NO	LOW CELL LIMIT	HI CELL LIMIT	RELA FREQ
1	0.	1.	.86
2	1.	2.	.1
3	2.	3.	.04

1 - Material
2 - Labor
3 - Inventory

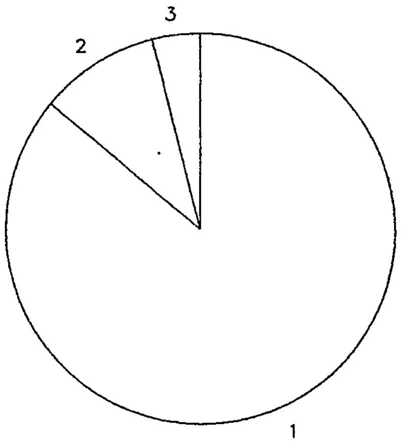


Figure 7. Single Stack Baseline Cost Breakdown.

Table 3. Optimized Order Selection Analysis and Cost Breakdown.

Run Description	$/Part	Throughput (parts/hr)	Average Utilization
Baseline	3.03	64.45	.5881
Baseline with Optimization	2.91	66.49	.5864
Best Case	2.70	61.00	.6452
Best Case with Optimization	2.72	58.72	.6510

Run Description	Material $/Part	Labor $/Part	Inventory $/Part	Total $/Part
Baseline	2.60	0.29	0.15	3.03
Baseline with Optimization	2.49	0.28	0.14	2.91
Best Case	2.27	0.30	0.13	2.70
Best Case with Optimization	2.28	0.31	0.13	2.72

Maximize

$$Z = \sum_{i=1}^{N} a_i x_i$$

s.t.

$$\sum_{i=1}^{N} a_i x_i \leq b$$

$$x_i = 0 \text{ or } 1$$

Where: a_i = area required to produce order i on the given stack type

x_i = the inclusion of the i order
0 if order i not included
1 if order i included

N = the number of orders in the critical part family file

b = the area available for the given stack type

Figure 8. Zero-One Formulation of Order Selection

Minimize

$$Z = b - \left(\sum_{i=1}^{N} a_i x_i \right)$$

s.t. None

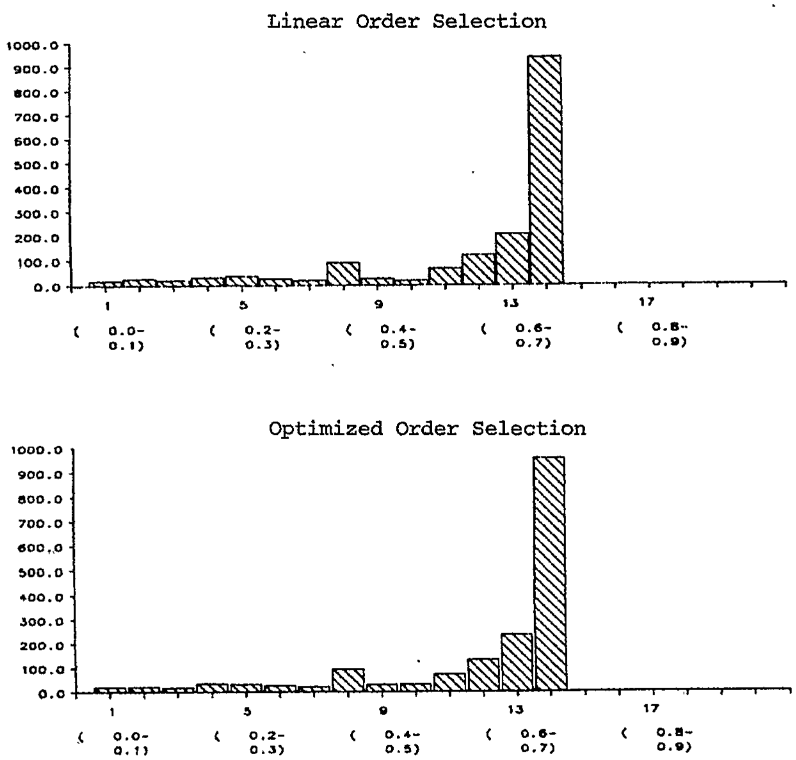
Where: a_i = area required to produce order i on the given stack type

x_i = the inclusion of the i order
0 if order i not included
1 if order i included

N = the number of orders in the critical part family file

b = the area available for the given stack type

Figure 9. Zero-One Formulation of Order Selection



Linear Order Selection

Optimized Order Selection

Figure 10. Material Utilization Distribution.

support software for decision making can be evaluated through simulation and the effect of changing various system parameters can be evaluated.

ACKNOWLEDGEMENTS

The authors would like to recognize the following persons for their input, ideas and concepts that were included in this paper: A. Alan B. Pritsker, Floyd H. Grant III, and Chuck Reilly.

REFERENCES

Booz, Allen and Hamilton, Inc., Sheet Metal Fabrication Systems. Booz, Allen and Hamilton, Inc., 8800 Pleasant Valley Road, Cleveland, OH 44131. (Preliminary Draft)

Pritsker, A.A.B. and C.D. Pegden. Introduction to Simulation and SLAM. John Wiley & Sons, New York and Systems Publishing Corporation, West Lafayette, IN 47906, 1979.

Standridge, Charles R. The Simulation Data Language (SDL) Reference Manual. Pritsker & Associates, Inc., P.O. Box 2413, West Lafayette, IN 47906, 1980.