INTEGRATED CIRCUIT SHOP SIMULATION

FOR CAPACITY PLANNING AND SCHEDULING

Dennis D. Newhart
Graduate Student-Industrial Engineering-Lehigh University
Development Engineer
Western Electric Corporate Education Center
P. O. Box 900, Princeton, New Jersey 08540

advised by
John W. O'Leary
Member of Research Staff
Western Electric Research Center
P. O. Box 900, Princeton, New Jersey 08540

SUMMARY

A large flow shop with cycling has been modeled with GASP II in an effort to study the effects of scheduling rules and capacity constraints. Consisting of multiple codes requiring 150-175 operations on 55 facilities, the shop, an integrated circuit line, was previously simulated in Fortran IV for the express purpose of determining the frequency of input messages to a shop information system. However, the Fortran IV model did not possess enough flexibility or efficiency for doing further analysis. By utilizing GASP II and its list processing techniques, shorter running times and implementations of scheduling algorithms could be achieved in a compact manner. The GASP II model is also capable of handling rework loops, yield distributions for each facility, priority assignments, and multiple channels per facility. Ultimately the model will have application for production scheduling in an on-line, real-time shop information system.

INTRODUCTION

Shop management proposed the installation of an on-line, real-time information system for a complex integrated circuit line involving nearly 175 operations and 30 product codes. The system was designed to provide each operator with a small, personal input station for entering data to a pair of processing computers. As lots passed through the shop, operators would be required to send messages at both the start and finish of an operation. These messages would be verified by the system which would use the data to keep a current status of all lots and their progress in the shop. Any events that were abnormal to specified shop procedures would be reported on an exception basis.

Before implementing the system in the integrated circuit shop, management needed to know more information about the traffic density of in-process lots. The final design of the system's hardware and software would depend on the quantity and the frequency of input messages that would be generated by movement of lots on the shop floor. To predict the anticipated density of messages, a simulation model was developed to aid in these design decisions. This model called SIMP was successfully run to produce stochastic sequences of input messages that were representative of the integrated circuit shop. These sequences, identifying the shop time of day and the associated number of messages, were in turn tested on the information system while it was still in the development laboratory.*

*The development of SIMP was contracted by the shop management with an outside group of system programmers.

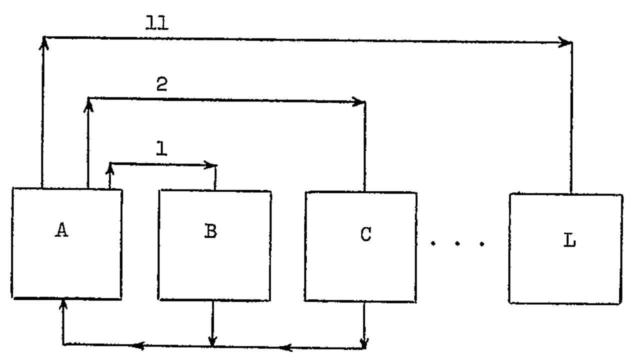
Realizing the insight to be gained through simulation models, management requested a feasibility study of SIMP's capabilities in evaluating capacity constraints and scheduling algorithms. The study revealed several negative characteristics of SIMP which will be discussed later in detail. In short, SIMP was inefficient and inflexible for redesign in relation to these new requests by management. Subsequently, a decision was made to construct a new model to be as efficient and as powerful a tool as possible.

Essentially this was accomplished by remodeling the shop to provide maximum ease in programming. Significant differences were achieved in computer running time and the amount of core required by the two models.

The details of the shop structure, the new model's characteristics, the essential differences between the models, and applications of the new model to shop problems will be emphasized in the following sections.

SHOP STRUCTURE

The integrated circuit shop can be characterized as a flow shop with cycling. The reason is that each product code follows a predefined path; and during one trip through the shop, each job will cycle or return a total of 11 times on a specified group of six facilities. This cycling feature is shown below in Figure 1.



Block A represents the six cycling facilities that will be revisited 11 times throughout the manufacturing of each job or lot.

FIGURE 1

For each different facility in the shop, there may be more than one machine or channel having identical processing capabilities. This homogeneous group of machines or channels will be referred to as a facility center. In total, the shop has 55 distinct facility centers with several of them used more than once in the entire processing of a given lot.

In addition to the cycling feature, rework on some lots may be necessary throughout the process. For example, after certain inspection points, a decision may be made to rework that percent of the lot which failed to meet testing criteria. Until this rework is completed, the parent lot (main lot) is held in abeyance at the storage point where eventually the offspring joins the parent and the two again proceed as one lot. It should be noted that for any one of the 175 operations the processing time typically consists of two major components. One time factor is a machine time which is independent of the number of units per lot. The other component is a processing time which is proportional to the number of units per lot. At specified points or stages in the integrated circuit line, yields are experienced by each lot which either fails to meet engineering specifications or receives improper handling by operators. The distribution of yields vary depending upon the completed stage of processing. Since one component of processing time is linearly related to the number of units per lot, the yields thereby add to the already stochastic nature of the total processing time per lot. Finally the line is run three shifts per day with the third shift workers performing only a few selected operations.

The size of the shop and the combination of all of the above features present a formidable problem in designing a model which can operate effectively and economically in aiding management decisions.

MODEL DESIGN

To describe the shop structure, the new model was constructed in the following manner. First each job is assigned seven attributes to define its status at any time during the simulation run.

Attributes

1. Scheduled Event Time;
2. Event Code;
3. Lot Starting Time in the Shop (A unique number);
4. Current Number of Units;
5. Current Position in Process;
6. Code Designation;
7. Priority Index for Scheduling and Ranking on Queues.

Secondly, for each possible routing or path to be followed through the shop, a sequence of facility numbers is defined to indicate the correct processing facility for a given operation. A quick study of the sequence below again reveals the cycling nature of the shop.

Example: Cycling on facilities 2, 3, 4
(1,<u>2,3,4</u>,5,<u>2,3,4</u>,6,7,8,<u>2,3,4</u>,9,...)

Basically the model is not oriented toward operation number but toward facility number. Furthermore, only one corresponding queue file is designated for each facility center. This assumption implies that all channels for a specific center select future jobs from a common queue as shown in Figure 2.

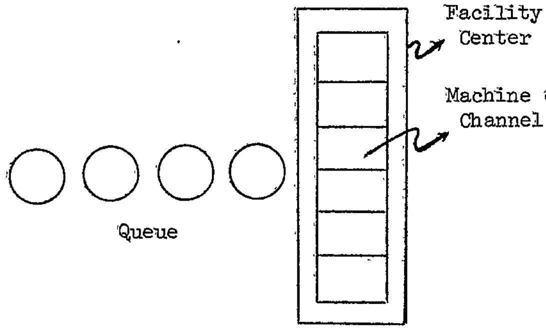


FIGURE 2

Also rework loops must be accounted for in each routing. This is accomplished by inserting a 0 facility number to designate the beginning of a rework loop and the correct rework facility numbers followed by a 99 to signal the end of this loop as demonstrated below.

Facility numbers: 1 ⟶ 4 ⟶ 5 ⟶ 3 ⟶ 7 ⟶

Rework loop required after facility number 5.

Routing: (1,4,5,0*,4,8,99**,3,7,...)

* Decision made after processing at facility 5 to either:

(1) Go to facility 3 for further processing or

(2) Reprocess part of the main lot on facilities 4 and 8.

** Arbitrarily high number for end of rework test.

Whenever a lot encounters a 0 facility number, a probabilistic decision is made whether or not to rework a certain percentage of the main lot. If a lot does not require rework, a test sends the entire lot to the facility number following number 99. It should be noted that an additional file must be created for parent lots when rework should be performed on the offspring lot.

The key to moving each job through the shop depends upon attribute 5 which indicates the current position of the job in the list of facility numbers. As a job finishes an operation, attribute 5 is increased by one to locate the next facility necessary for processing. As mentioned above, if all channels at a facility center are busy, then the job waits in the corresponding queue file. When the job finishes the last process step, the final lot status is put on disk storage where the data by lot can be used at a later date for further statistical analysis.

One last addition to the model entails an event to shut down some shop facilities on third shift. Whenever the model enters a third shift time period, designated facilities are declared inoperative. Then at the start of the first shift, these facilities are again returned to normal operative status.

PROGRAM FEATURES

The new model is programmed in GASP II. To handle the large number of events occurring in a simulation, GASP II utilizes the principles of list processing. In ordering a file, list processing does not demand that each record be placed physically in its correct position. Instead, each record has two pointers with one indicating the row number of the logical preceding record and the other showing the row number of the logical succeeding record. This method allows a new record to be filed in any empty row as long as its pointers indicate its relative position to the other records. At the expense of the additional core required for the pointers, list processing is significantly faster than the conventional "Search and Pushdown" methods when the number of records being ordered is very large.

Looking at the integrated circuit shop with about 200 in-process lots and each one requiring nearly 175 operations, one finds a sizable number of events occurring close together in time. SIMP's program cannot effectively cope with this problem since all future events are ordered by type in four different files by means of a "Search and Pushdown" method. Also, before an event can be selected, a comparison has to be made to determine which one of the four files contains the event closest to the current simulation clock. In contrast the new model in GASP II employs only one event file ordered by list processing. The consequences of SIMP's programming design are revealed by its excessive amounts of computer time needed to run the model.

Another disadvantage of SIMP is the difficulty in preloading the facility queues with lots before starting the simulation. This fact coupled with the long time required by SIMP to reach steady state impedes a decision maker in economically performing a capacity analysis for this shop. For this analysis, many alternative configurations of machine center sizes must be evaluated, thereby increasing the desirability to preload the shop in an approximate steady-state condition and eliminating needless computer time required for stabilizing shop loading. For models smaller than the integrated circuit shop, these conclusions will not always be true.

Finally SIMP does not allow adequate provision for interfacing a scheduling algorithm subroutine. Facility center queues must be ranked and evaluated as time progresses in the simulation so that rules like shortest processing time or COVERT can be properly evaluated by the shop management.

This problem of scheduling can be easily solved by features of GASP II used in the new model. First each queue may be ranked according to any job attribute including a priority index. Furthermore, the ability to reference each member of the queue, the number in the queue, and the waiting times in the queue allows many types of dynamic algorithms to be developed in selecting a job or lot from a queue. Another powerful method in selecting records from files is a GASP II subroutine called FIND. This routine will locate that record in the file whose attribute satisfies some reference value in question. This feature is useful not only in scheduling but also in recombining parent and offspring lots at the end of a rework cycle.

In viewing the question of how often should queues be ordered during the day to provide effective scheduling, one need only supply a periodic event in the model to activate the scheduling algorithm at desired intervals throughout the simulation. This exogenous event capability is further utilized by dividing the shop day into 8-hour events to determine when certain facilities should be shut down on third shift or start processing on first shift.

A comparison of SIMP and the new model in Table 1 shows the differences that can be chiefly attributed to a model design that fits the identity of the shop and a concise programming effort using GASP II programming techniques.

	SIMP	NEW MODEL
Manpower Required for Development	30 man months	2.5 man months
Time to Reach Steady State	10 hours	0.67 hours
Time to Simulate One Shop Day	8.3 min.	0.60 min.
Core Required on IBM 360	200 K	70 K

TABLE 1

SIMP was developed by a group of system programmers who relied on their forte of constructing sound, logical programs. Consequently SIMP contains many nested Do loops, nested matrices, and considerable file manipulations. The final result resembles a logic program lacking in simulation techniques. Unfortunately this approach in modeling large shops is inefficient, and it eliminates any feasible analysis of shop problem areas other than the generation of input messages.

VALIDATION

Once steady-state conditions were reached at 2000 hours with a fixed input rate and starting lot size, several shop characteristics were measured for an additional 4000 hours. These characteristics included:

1. The average number of lots in-process;
2. The average number of units in a finished lot;
3. The average throughput time for a lot;
4. The average size of a queue at a facility center;
5. The average waiting time in a queue;
6. The average utilization of a facility center.

These statistics were reviewed and favorably accepted by the shop operating personnel as being a realistic picture of the shop. Furthermore, a shop information system operating with porta-punch cards on a batch-mode basis additionally confirmed many of the model's statistics for the current set of conditions. Other historical conditions with larger starting lot sizes and greater operating capacities at certain facility centers were validated under the same procedure described above.

MODEL APPLICATIONS

The immediate result of the data in Table 1 indicates that the new model can now be put to effective use in studying both scheduling algorithms and capacity constraints. First an evaluation of both static and dynamic scheduling rules can be made to determine which rule is optimal according to the criteria of the integrated circuit shop management. The rules to be tested will vary in complexity from first-come first-served to screening and optimum-seeking search techniques.[2] Then to determine the "best" scheduling rule, an analysis of the tradeoff to shop performance and the associated computational time for each rule will be required to prevent the system from spending too much time on scheduling and not enough on its basic duties. A hypothetical graph of these relationships is shown below in Figure 3.

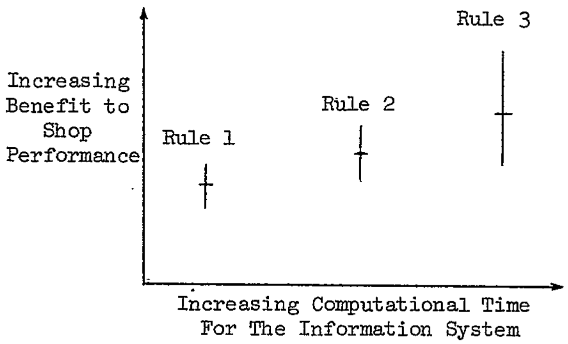


FIGURE 3

The end result will allow shop operators to request at any time an ordered list of all jobs awaiting processing at their facility center. However, other alternatives exist for applying the algorithm. Perhaps all queues may be ordered once a day on third shift when the system is relatively idle, and this may still provide effective scheduling. All of these possibilities will be tested on the new simulator.

The other area of capacity analysis holds great potential for savings. Major investment decisions for new equipment can be further evaluated by analyzing the shop performance under current and future loading conditions. Analytical methods cannot possibly account for the complex interactions in the shop operations. It is hoped that the new model will help to prevent decisions that tend to suboptimize one section of the line. Finally many other problems will be open to investigation such as machine breakdowns and the optimal number of maintenance workers required throughout the three shifts.

CONCLUSION

The accomplishment of building an efficient, compact model highlights the concept that structuring a simulation model can be critical and programming techniques contribute greatly to the overall model performance. The new model has many facets which added to its success. The principal ones are listed below.

1. The model is primarily oriented toward facility number instead of operation number.
2. Maximum use is made of the seven job attributes.
3. List processing techniques aid in ordering the large files.
4. Special GASP II subroutines contribute to the ease of filing and locating records.
5. The event concept is employed to describe shop policies.

Building and designing a simulation model demands the recognition and utilization of many concepts discussed in this paper. If these ideas are adhered to by model builders, they will produce a responsive, flexible model capable of economically solving a wide-range of problem areas.

BIBLIOGRAPHY

1. Anderson, J. A., "A Study of Dispatching in a Production System Characterized by Cycling," Unpublished Master's Thesis, Lehigh University, 1971.
2. Emery, J. C., "Job Shop Scheduling by Means of Simulation and an Optimum-Seeking Search," Proceedings of the Third Conference on Applications of Simulation, 1969.
3. Pritsker, A. Alan B.; and Kiviat, Philip J., Simulation with GASP II, Prentice Hall, Inc., 1969.