Proceedings of the 1985 Winter Simulation Conference
D. Gantz, G. Blais, S. Solomon (eds.)

IMPROVING A MULTISTAGE/MULTIPROCESSOR FLOW-SHOP PROBLEM OF NUMEROUS TECHNOLOGICAL CONSTRAINTS THROUGH SCHEDULING

Way Kuo, Jon Yanney, Russell Tsai
Department of Industrial Engineering
Iowa State University
Ames, Iowa 50011

ABSTRACT

The manufacturing of industrial rubber compounds is a three-stage process on a Banbury production system. At each stage the rubber is mixed (in a Banbury mixer) and transferred to next-stage storage. Depending on requirements, the first two stages may be repeated up to three times. Finally, rubber is mixed on the third stage and then milled to obtain a slab form of the new compound rubber.

The objective is to derive a "good" heuristic scheduling algorithm that uses a series of sorts, searches, and simulations in order to sequence and route jobs through production stages. Evaluation criteria are based on the increase in equipment utilization and on satisfying quality and operating restrictions. Schedule feasibility will be tested by a GPSS simulation model of the production system.

Key Words: Banbury Operation, simulation, scheduling

1. INTRODUCTION

In the rubber tire industry, the first manufacturing step is the production of various rubber compounds used as input materials for tire subassembly parts. A tire consists of anywhere from 5 to as many as 15 distinctly different rubber compounds. Each compound offers a variety of physical and esthetic properties.

These rubber compounds are mixed in discrete batch runs on a processor known as a Banbury Mixing Chamber. Often the mixing of a rubber compound constitutes two or more individual processing stages. Consequently, "production lines" of Banburys, including a mixer, additional processing equipment, material handling equipment, and intermediate storage facilities, are configured similarly to a production line for a discrete metallic product. Because of the extreme importance of quality in the production of rubber compounds, quality restrictions are the most critical parameter of the production system. Therefore, in scheduling terminology, a Banbury production system is of the flowshop type where the technological constraints suggest an order of processing between first-stage processing, second-stage processing, and so on. It is also a system governed by numerous restrictions on the technological constraints including processor preferences and the splitting of jobs over multiple processors. The scheduling problem it represents is considered NP-complete [1,5] and is of a highly complicated n/3/F/Cmax type.

This study provides an application of a feasible and practical heuristic scheduling algorithm that statically sequences and routes rubber compound jobs through a multistage/multiprocessor Banbury production system. The algorithm provides a good solution and assures optimal equipment use relative to quality and operating restrictions.

This paper presents the development process of the scheduling algorithm. First, the modeled Banbury production system and its associated operating and quality restrictions are described to familiarize the reader with the scheduling problem. A brief review of the literature is then discussed, leading to the objectives and assumptions governing the investigation. A GPSS simulation model is then detailed along with the results of the effectiveness of some very simple heuristic scheduling rules applied to the scheduling problem. Finally, the development of the actual heuristic algorithm created for this unique problem is described.

2. BANBURY PRODUCTION SYSTEM

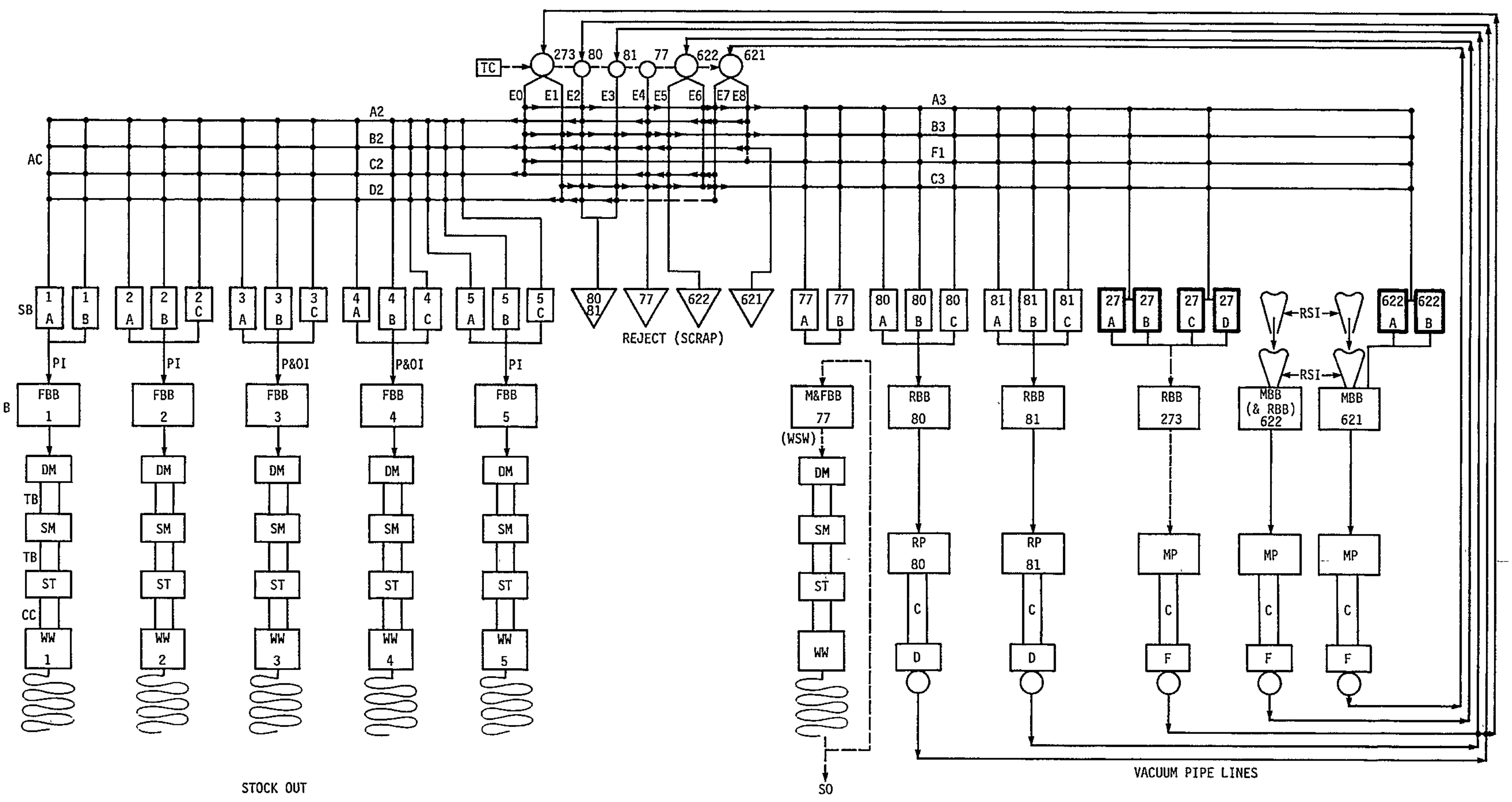
A typical Banbury Department, or Department 112, consists of eight Banbury production lines (Figure 1). The MB621 and MB622 lines are dedicated to the initial mixing of rubber compound constituents known as the masterbatch stage. The RB273 line is dedicated to the breaking down of rubber polymer chains, which is known as the remilling stage. The FB71, FB72, FB73, FB74, and FB75 lines are dedicated to the final mixing of rubber compound constituents; this is known as the final stage. Either of the first two stages may be repeated depending on the processing requirements of the rubber compound.

Department 112 produces approximately 125 different stocks annually, 75 of which are produced at least once every two weeks. Each time a stock is run, from 7 to 42 masterbatches at approximately 1020 lbs per batch are produced.

When a stock is masterbatched, a variety of raw ingredients are both manually loaded onto a conveyor belt that empties into the mixer and automatically injected by computer-controlled equipment into the mixer. The batch is then mixed for 1.0 to 2.5 minutes. After being mixed, the batch is pelletized, lubricated, dried, and transferred to storage bins.

The next stage for a "one remill" stock is remill processing, a stage in which no new ingredients are introduced but further breakdown of the batch is achieved by added mixing time. Approximately 1020 lbs of pellets are drawn from one of four storage bins and mixed for almost 2.5 minutes. The batch is then dropped into the remill pelletizer, where it follows a course of processing and transporting similar to the masterbatch line until it reaches a storage bin above a final Banbury mixer.

BANBURY SYSTEM DIAGRAM



C = DEWATERING CONVEYOR
D = DRYER
F = FESTOON DRYING SYSTEM
MBB = MASTER BATCH BANBURY
RBB = REMILL BATCH BANBURY
FBB = FINAL BATCH BANBURY
MP = MASTER BATCH PELLETIZERS

RP = REMILL BATCH PELLETIZERS
DM = DISCHARGE MILL
SM = SLAB-OFF MILL
ST = SOAP TANK
WW = WIG-WAG MACHINES
PI = PIGMENT INPUT
P&OI = PIGMENT & OIL INPUT
RSI = RAW STOCK INPUT

SO = STOCK OUT
TC = TOWER CONTROL
PR = PELLET RECEIVERS
B = BANBURY
SB = STORAGE BINS
AC = AUGER CONVEYORS
TB = TAKEAWAY BELT
CC = COOLING CONVEYOR

CAPACITIES
BANBURIES

MBB
≅ 1050#/BATCH ≅ 2.2 min/cycle

RBB
≅ 1050#/BATCH ≅ 2.2 min/cycle

FBB
≅ 400#/BATCH ≅ 2.6 min/cycle

STORAGE BINS:

☐ 7,000#

☐ 14,000#

▣ 19,000#

Figure 1: Banbury system diagram

Final mixing takes place on one of five smaller Banburys. The batch capacity for each is approximately 400 to 460 lbs, depending on the rubber compound. In this stage, final mix ingredients are introduced into the mixing chamber from the final storage bins, an offloading conveyor, and by automatic injection. A batch is mixed for 2.0 to 2.5 minutes, depending on the heating temperature requirements. Once mixed, the batch is milled into a 3-ft by 100-ft sheet of rubber. The rubber compound is stacked and tested for quality. If it passes the quality test it then becomes available for tire production.

Currently, a random approach for scheduling jobs through the system is used. The production demand requirements listing, which suggests an order of job processing, is provided to Department 112. Supervisors then determined a sequence of jobs to be processed for each of the two masterbatch Banburys. Routing of the jobs through the system is determined by a material handling operator who directs stocks into the next-stage bins on the basis of the storage bins' current status. Routing decisions are made as the job arrives from the first-stage processing.

2.1. OPERATING RESTRICTIONS

The complexity of the operating restrictions is amplified by the fact that, because of continuous operations, there are always current production parameters that must be considered. Therefore, in terms of defining a static schedule, not only must new jobs be scheduled, but scheduling of current jobs must be completed first so that production parameters such as current storage capacities and busy machines can be defined.

Another set of restrictions, which we term flow rate restrictions, evolves because of the variation in mixing times. It has been determined that a large variation in mixing times occurs between numerous physical and conceptual components and their functional relationships (i.e., between batches, between processors, between stages, and between days).

Because of these variances, a discrepancy in net processing rates can occur between system input processing rates and system output processing rates, and between stage input processing rates and stage output processing rates. The problem of net negative production rates is negated by the use of large intermediate storage bins between stages; however, these have limited capacities. Therefore, steady-state system processing often results in machines going down because of the lack of output storage.

Another set of restrictions exist which are stock dependent and have a great effect on the system processing efficiency. For unidentifiable reasons, certain stocks process up to 75% faster on some processors. These efficiency processor preferences need to be considered during scheduling.

2.2. QUALITY RESTRICTIONS

Numerous quality restrictions are associated with the production system. One is that not all stocks can be run on all Banburys because of ingredient injection limitations and because some Banbury lines process certain stocks more effectively. Another restriction is that strings of jobs often must be processed to provide a method of cleaning the production line for critical stocks. A third restriction is that some stocks require repeating a processing stage in order to attain homogeneity or to receive more raw ingredients.

2.3. CONCERNS

Scheduling the Banbury production system poses an immensely complex problem because of the following system characteristics:

- Repeatable stages within the three stages
- Multiple processors within stages
- Unpredictable flow rates
- Many quality restrictions and technological constraints
- Existence of current production parameters (system is always partially full)
- Splitting of jobs over multiple processors
- Processor preferences due to quality and operating restrictions
- Frequently scheduled and unscheduled maintenance of equipment

A review of scheduling literature suggests that theoretically based, constructive scheduling algorithms are not used in an attempt to solve problems of such complexity [3]. To solve a flowshop system scenario optimally with numbers of machines (or stages) greater than two generally requires the evaluation of complete enumeration of routing possibilities or a sophisticated branch-and-bound approach. In both cases the computational requirements, both time and power, are extensive when the number of data required to define the particular problem is greater than 30 [4]. It is therefore assumed that the additional system characteristics would increase computational requirements so that the problem could not be solved in a timely fashion even if an optimal solution existed and an algorithm could evaluate such a problem.

2.4. ASSUMPTIONS AND OBJECTIVES

Based on the previous research findings, our intent is to produce a heuristic algorithm that sequences and routes jobs through the system while taking into account the various complexities unique to the system. Simplifying the system characteristics by creating restrictive assumptions will be minimized. This will give the algorithm more credibility and increase the likelihood of implementation because of its ability to deal with the characteristics. The resulting schedule will be used as a tool by the scheduler to aid in decision making. The overall objective of the research is to produce a practical algorithm that will suggest a "good" viable solution as often as possible.

The assumptions follow:

1. All mix times are to be considered constant for computation purposes.
2. Stocks, with their number of remills dependent on mixing quality, will have a constant number of remills for static scheduling.
3. Jobs currently in the system but not bound to complete routing through the system will be routed with respect to the new set of jobs to be scheduled.
4. If a "good" schedule cannot be determined within a reasonable amount of time, the algorithm will terminate and the scheduler will be requested to complete the scheduling process.

Output from the scheduling algorithm will specify the sequencing of jobs as they arrive at each processor as well as the percentage of a job to be processed, and the designated routing to the next stage for that percentage of a job will be given.

The scheduling package as a whole needs to be designed using well-defined resources that satisfy certain department needs and criteria. The amount of work time that can be committed to the scheduling process is minimal. A small personal computer (256-K RAM, 1 floppy disk drive, 1 10MB hard drive) and printer were allocated to the project. Because of the high failure rate of equipment within the system (total system effectiveness in terms of equipment utilization ≅ 65%), the scheduling process needs to be executed briefly and immediately preceding the scheduled run. This is necessary in order to quantify current production parameters accurately at the time of the scheduled run.

The goal of the scheduling project is to complete the scheduling processes in less than 45 minutes. This includes the time to collect and enter into the computer all production demand information and current production parameters, the time for actual schedule processing, and the time for printing job sequencing and routing listings for various operators and supervisors.

3. THE SIMULATION MODEL

In order to perform some preliminary analyses and to test the feasibility of the resultant schedule, a GPSS simulation model was developed [8]. This model allowed us to evaluate several heuristic scheduling approaches. It was also instrumental in determining several flowrate formulas. Finally, it was used to determine if the schedule produced by the adopted algorithm was in fact a viable schedule which would direct the flow of jobs through the system without overflowing storage capacities and flooding the second and third stages.

Because of the complexity of the Banbury system, a simplified model of the system was adopted first. Details were then added to this model until the responses of interest were accurately represented. The following describes the simulation inputs, the model structure, and the output.

3.1. INPUT REQUIREMENTS

The input data necessary to drive the GPSS model reside in a job file. The file describes the entire set of jobs and the specifications of each individual job.

For each job, 17 variables--number of masterbatches, batch weight in each stage, etc.--have to be set up before the simulation. A FORTRAN subroutine has been written to communicate with the GPSS program. Job specifications are stored in a data cell that can be read by the FORTRAN subroutine. In this way the data can be initialized efficiently. More importantly, the job sequence can be reordered easily to simulate various dispatching rules. The control statement CALL and the block BCALL, which provide an interface to allow the FORTRAN subroutine to be called from a GPSS/H model, are employed to perform this task. The total number of jobs to be simulated and the simulation time are read through the CALL statement before the start of the simulation. Job specification is read through the BCALL block whenever a job transaction is generated.

Since the job specification is provided by calling a FORTRAN subroutine, the developed simulation model can accommodate the variation of batch weights and mixing times. This can also encourage non-GPSS users to apply this simulation model as a tool for decision making.

3.2. MODEL STRUCTURE

The following five segments have been constructed in the GPSS simulation model:

- Masterbatch Banbury processing segment
- Stage (routing) selection segment
- First-stage (RB622) remilling segment
- Second-stage (RB273) remilling segment
- Final Banbury processing segment

Figure 2 illustrates these five segments and their relationships. In this model, all jobs are generated at the same time; i.e., all jobs are available at time zero. As soon as a job enters the BCALL block, the FORTRAN subroutine is called. The job specification information is then assigned to a half-word matrix whose first subscript represents the job number. According to the value representing the number of masterbatches to be produced, the job is split into the appropriate number of transactions representing the masterbatches. After a batch is processed through the masterbatch Banbury segment, the transaction is routed to the stage selection segment to determine its successive processing segment. If the batch requires a first-stage remill, it will be routed to the first-stage remilling segment. Then it will be routed back to the stage selection segment again for further processing. If the batch is a straight stock (no remill required), it will be routed directly to the final Banbury segment and then leave the model.

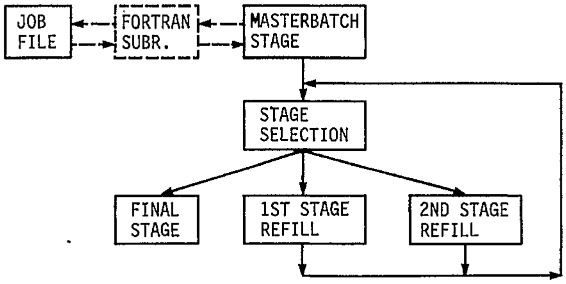


Figure 2: Model segment relationship

Since the final batch weight and masterbatch weight are quite different, care must be taken to split the masterbatch transaction into the appropriate number of transactions representing final batches. First the masterbatch weight is divided by the final batch weight, with the number of new transactions representing the integer portion of the resulting value. Then the remainder of the division is accumulated and stored in a matrix. The accumulated value stored in the matrix is added to the next masterbatch transaction arriving at the block.

In this Banbury system, each individual batch run is continuously processed while the batch runs are generated at discrete intervals. Because of the GPSS capability to divide a batch transaction into two or more sub-batch transactions, it allows split batches to be evaluated dynamically. Thus, success has been achieved using the discrete-event simulation language to simulate a continuous processing system [6].

3.3. OUTPUT

In addition to the standard GPSS output report, the flowtime and number of final batches produced for each job are also stored in a half-word matrix. Therefore, after every simulation run the following statistics of interest are collected:

- utilization of each Banbury mixer
- queue statistics of each storage bin
- processing routing for each job
- total number of final batches produced for each job
- total flowtime for each job

It must be kept in mind that when explaining and analyzing the simulation result, the final representation of a transaction is the final batch.

3.4. SCHEDULING RULES

Actual production demand lists and average current production parameters are entered into the scheduling package, and a schedule is obtained. Scheduling rules are given in the Appendix. This schedule of the sequencing and routing of jobs through the system is stored as data in the job file of the GPSS simulation program. The simulation is run, and queue statistics are analyzed. Any waiting time in a storage bin queue for the first batch of a job is considered a mixing of jobs. (A storage bin becomes available when the last batch of a job is depleted and the gate block reset.) Any waiting time in a storage bin queue for any batch (after the first batch) is considered overflow.

4. SIMULATION RESULT

4.1. DATA

The GPSS model was employed in the investigation of some simple, conventional heuristic scheduling rules or techniques. All material handling and compound mixing times used in the simulation model were based on three month's data collection at the plant. All input jobs for simulation runs were obtained from actual job runs scheduled by the department. The use of actual data allowed for a more credible and viable result in the evaluation of these scheduling rules. All simulation runs were made using a GPSS/H processor installed on an Itel AS/6 computer.

4.2. RESULT

The following tasks have been performed using the developed simulation model:

System Figures:

- altering Banbury mixing time in each Banbury stage respectively to study the bin depleting rate
- altering the number of final Banburys to study the machine utilization and queue status of the bins

Dispatching rules include:

- SPT (shortest processing time)
  - determining the job sequence according to processing time without considering straight stocks or remill stocks
  - separating straight stocks and remill stocks to two classes. Each class uses the SPT rule; then they are interlaced with each other.
- MST (minimum slack time)

  scheduling job sequence according to non-decreasing slack times, then performing a and b as described above. In this study, the due date is assumed 16 hours in advance for all jobs.
- running short stocks (required smaller number of batches) on one of the two masterbatch Banburys, and running long stock (required larger number of batches) on another Banbury

  The total number of batches to be produced on each Banbury mixer is assigned to be approximately equal. The job sequence in each machine is determined by the SPT rule.

Ten distinct job sets with different job specifications were simulated for evaluation of each of the five scheduling rules. Each job set consists of 12 to 15 jobs and represents one shift's work. The simulation result of the random scheduling rule is used as the criterion to measure the improvement of these five simple scheduling rules. The resulting statistics of maximum flowtime and mean flowtime are displayed in Table 1. For each category of the schedule rules in the table, the first row shows the values of flowtimes and the second row shows the percentages of improvements when compared with the random scheduling rule.

Table 1. Summary of simulation result

Rules	Flowtime (minutes)	
	Maximum Flowtime	Mean Flowtime
Random	637 -	372 -
SPT (processing time only)	658 -3%	330 11%
SPT (with interlacing)	635 0.3%	328 12%
MST (processing time only)	728 -14%	423 -14%
MST (interlacing)	606 5%	393 -5.6%
Short stocks vs. long stocks	649 -2%	330 11%

Considering maximum flowtime, the percentage of improvement is not significant. Among these scheduling rules, the MST rule with stock interlacing is the best because it reduces the maximum flowtime by 5%. For mean flowtime, the SPT rule with stock interlacing improves the statistic by 12% without deteriorating the maximum flowtime.

Minimum mean flowtime will result in minimum average number of jobs in the system. Thus, the mean flowtime is directly proportional to the mean work-in-process inventory as measured by the number of jobs [2]. In addition, since the plant is run continuously (three shifts a day, seven days a week), the improvement of the mean flowtime is more important than that of the maximum flowtime. Based on this criterion, the SPT rule with stock interlacing has served as the starting point for the scheduling package. This has reduced computer time needed to reach a satisfactory solution.

5. CONCLUSIONS

The scheduling package is currently being implemented. Thus, while actual package success cannot be documented, several benefits are definitely expected.

One is that the implementation of the scheduling package is already instilling a greater awareness of production status. From this alone the number of incidences where jobs are mixed are expected to decrease drastically.

Another point is that critical or late stocks are expected to leave the system in an expeditious manner. Before the package was implemented, some jobs were lost at the shift change because of poor communication.

A third point is that operator morale is expected to improve because certain runtime considerations were added to the algorithm. For example, a heavy clay stock, usually a very dirty stock to load, will not be scheduled after workers have showered.

The resources required to support the package are considered minimal compared to the expected benefits. Apart from the hardware support, one entry operator will be required to collect current production parameters (5-7 minutes) and then enter that data along with the production demand into the computer (5-8 minutes). The schedule is then calculated and printed (15-20 minutes). A schedule is required during each shift of production for three shifts daily.

Obviously, the resulting schedule is not optimal. Nor, from a practical standpoint, can an optimal schedule be determined. This is because subjective scheduling rules are introduced into the algorithm.

In cases where third-stage equipment is failing frequently because of high humidity and sticky rubber, it is probably safer to split critical stocks over many final processors so that one major breakdown does not prevent stock from reaching the production floor. In some instances clay stocks should be run as the last job, albeit at the expense of worker morale and possible worker slowdown. To define true optimality would require a well-defined weighting scheme of conventional evaluation criteria (C_{max}, L_{max}, etc.) and subjective scheduling rule values.

Complete test results are not yet available. However, based on preliminary results, the algorithm is expected to produce a good, practical schedule at least 75-80% of the time. This figure could increase if an initial screening of the production demand test was made by the supervisor. (If a problem area is found, it can generally be rectified with the substitution of one or two jobs.) The GPSS simulation model is used to verify the feasibility and practicality of the schedule. A schedule is designated as feasible if the GPSS simulation does not reveal any overflowing of storage bins or mixing of jobs in the storage bins.

6. ACKNOWLEDGMENT

This study is partially supported by the Firestone Tire and Rubber Company, Des Moines Plant. Support through the Engineering Research Institute and the University Research Grant Program of Iowa State University are also acknowledged.

7. APPENDIX: DESCRIPTION OF THE SCHEDULING ALGORITHM

The preliminary analysis proved useful in developing a heuristic scheduling algorithm. It reaffirmed the initial logical assumption that interlacing straight stocks (no remill stage) with remill stocks (requiring at least one remill stage) was in fact a good sequencing method. However, to sequence and route jobs through the system requires a much more complicated set of scheduling rules.

The following outlines the scheduling algorithm developed for Department 112. First a brief description of the supporting data base is given. Then the actual algorithm is highlighted. Not all of the conditional tests included in the decision-making process are detailed. Rather, the philosophy of the scheduling process is emphasized by describing and justifying various procedures. Finally, the output from the algorithm is discussed.

In order to save data entry time, a data base of stock and system configuration information was created and stored in off-line memory. Constant data such as stock mixing times, specific gravities, and machine preferences were compiled and stored for each stock. System configuration information such as the number of final batch processors and storage bin capacities were also stored to make a more flexible and dynamic program. Dynamic scheduling data such as the stock names of jobs to be scheduled, the quantity of the each stock to be produced and the current status of storage capacities are obtained from the operator. Great care is taken to ensure the integrity of all input data. A series of verification procedures is executed in the early stages of the scheduling process so that entry errors can be rectified before too much time elapses.

Once all data required for scheduling a production shift are compiled, the scheduling process begins. Several preparation procedures are executed before the actual sequencing and routing steps begin. However, each is important and necessary to the scheduling process. The following description highlights all important procedures.

1. A total job processing time is computed for each job to be scheduled. The computation reflects the time the job would be in the system if it were processed first and if the system were clear.
2. All strings of jobs are found from the list of jobs to be scheduled. Each string becomes a large job and replaces the string of constituent jobs in the list. Each new string job is assigned the predominant or average job data attributes such as composite stage weights, specific gravities, and machine preferences.
3. A simulation of the processing of current stocks is performed in order to help quantify the "business" of the system. It also

identifies specifically which processors and storage bins are busy. In this manner, the initial flow of jobs to both the second and third stages can be regulated, busy processors can be avoided, and storage bins will not be flooded.

This simulation is accomplished in three steps. In the first step all jobs currently executing to final Banbury storage bins are allowed to fill those bins to capacity. Further processing past the second stage is stopped. The second step simulates continued production for jobs currently in the first two stages. Instead of filling the third stage final storage bins, job statistics such as remaining batches and third-stage arrival times are accumulated in a final Banbury queue matrix. It represents all jobs that are waiting for routing assignments into the third stage. These assignments are made later as they are determined when the needs of jobs to be scheduled have been analyzed. The third step completes the simulation processing of the third stage. Statistics on when bins and processors become available are collected.

4. A processing time-availability table is stored in matrix form for all processors. Each processor is allowed two "down" periods in which the processor is busy either processing a current job, being fixed under an unscheduled maintenance event, or being maintained under a projected scheduled maintenance action. All processor downtime information is entered by the operator. Storage bin downtime information is stored in a similar manner.

5. Once the times that processors become available for new job processing are determined, slack time computations are performed. Each stock's starting time restrictions, based on processor preference needs, each processor's earliest availability, and stock ingredient preparation times are analyzed and slack times determined. Also, because there are frequent equipment downtime occurrences, maintenance times on equipment are also considered. Figure 3 pictorially describes the slack time determination for a 1-remill stock with a MB621 processor and FB73 processor quality preference. The stock requires large quantities of clay and, for preparation purposes, must be run before the sixth hour of the eight-hour shift. The enlarged shaded areas represent available processing time. The final "good" processing time is projected in the slack time line.

6. The "business" of the system is broken down into nine levels. An analysis is performed by reviewing busy processors and the present stock weight in the storage bins. Then a level of business is assigned which suggests the direction of the initial flow of new jobs to either the second stage or the third stage. This is a critical factor and needs to be precisely defined in order to maintain high equipment utilization and high system throughput.

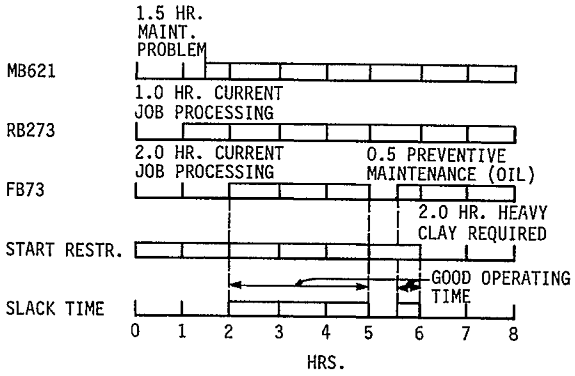


Figure 3: Slack time schematic for a 1-remill stock

7. At this point a concerted effort is made to separate the jobs to be scheduled into two queues, Q621A and Q622A. These two queues represent jobs selected by immediate processing requirements, processor preferences, and by straight stock-remill stock interlacing. Remill production is nearly equivalent to straight stock production; therefore, it is necessary that the one remill processor is kept busy in order to keep up with the demand.

   Separating the jobs into the two queues is performed in four steps. In step one, the list of jobs to be scheduled is separated into three queues. Jobs with an MB621 processor preference are assigned to Q621. Jobs with an MB622 processor preference are assigned to Q622. Jobs with no processor preference are assigned to Q620. In step two, the three queues, Q621, Q622, and Q620, are sorted first by the stock critical weight (a constant production attribute) and then by due-date weight (a dynamic binary demand attribute). The second sort is performed using a bubble sort to conserve the initial sort. In step three, Q621A and Q622A are constructed. All late jobs, termed "shift 0" jobs, from Q621 are added to Q621A. Likewise, "shift 0" jobs from Q622 are added to Q622A. Two passes are used to accomplish shift 0 additions. The first pass loads jobs with quality preferences, and the second pass loads jobs with efficiency preferences. The shift 0 jobs from Q620 are loaded into Q621A or Q622A in a manner that evenly weights processing loads. Once all late jobs are added, step 4 begins by adding jobs due at the end of the coming shift, termed "shift 1" jobs, using a similar method.

   When "shift 1" jobs are added, care is taken to load equivalent weights of straight stocks and remill stocks. It is noted that Q622A is heavily loaded with remill jobs. It was arbitrarily selected as the main supplier of remill stocks to RB273 in order to simplify the loading scheme.

8. A cleanup procedure is executed to make sure an equivalent loading of total processing time on each processor exists.

9. A series of four sorts is performed on the two queues to stratify jobs by (1) criticality weight, (2) late start prohibited, (3) early start prohibited, and finally, (4) shift weight. Once again, a bubble sort is used to preserve initial sorts.

10. Two new queues are constructed--Q621B and Q622B--which represent the final sequence for jobs awaiting processing on the MB621 and MB622 processors, respectively. A careful selection is made for the first job of each queue. It must satisfy the flow requirements and must try to meet due date restrictions. Several search passes are executed on Q621A and Q622A to accomplish this. If a first job candidate is not found in one queue, the other queue is searched for a job meeting specifications including quality preference restrictions.

11. After the first jobs have been added to Q621B and Q622B, the rest of the shift 0 jobs are located and fit into the queues. Each time a job is added to Q621B or Q622B, production statistics are kept on affected processors through the second stage. Storage capacities are checked and new values for "equipment available times" are reestablished. Once calculations on processing simulation have been tabulated, the next type of job--straight stock, remill stock, large or small job size, etc.--is sought.

12. Shift 1 jobs are added to Q621B and Q622B using the same search and computation techniques. These shift 1 jobs are added until a shift plus one-hour processing time has been loaded on each masterbatch processor.

13. As processing computations through the second stage are made, the Final Banbury Queue matrix is loaded with job statistics for third-stage processing (i.e., total weight of job awaiting final processing, arrival time to the final stage, etc.). This matrix of jobs awaiting routing through the third stage is now processed. Routing of these jobs is a critical process. Each job must be assigned to a particular Final Banbury (or a set of Final Banbury processors) and its corresponding storage bins. Tracking material flow through the storage bins is particularly critical from a quality standpoint. Each assignment is dependent on the current status of production on the equipment at the point of assignment and on the number of future jobs that require processing on the candidate equipment. It is noted that most jobs need to be split over two or more Final Banbury processors.

The output from the algorithm is brief and yet informative. The eight processor operators and the material handling operator responsible for material storage each receive a schedule specifically for their equipment. The schedule shows the sequence of stocks arriving and, for the latter stage operators, the bins from which they will draw their rubber. It also shows arrival times of jobs (or parts of jobs) to either designated processors or storage bins.

With this information, all personnel will know the precise order of processing and material handling events. More importantly, the next shift operators, upon arrival, will know the exact status of current production, including the correct order in which to deplete full bins.

8. REFERENCES

1. Cook, S. A., "The Complexity of Theorem Proving Procedures," Proc., 3rd Annual ACM Symp. on the Theory of Computing, 1971, pp. 151-158.

2. Conway, R. W., W. L. Maxwell, L. W. Miller, Theory of Scheduling, Addison-Wesley Publishing Co., Reading, Massachusetts, 1967.

3. French, S., Sequencing and Scheduling--An Introduction to the Mathematics of Job-Shop, N.Y.: Wiley, 1983.

4. Garey, M. R., Johnson, D. S., Computers and Contractability: A Guide to the Theory of NP-Completeness, San Francisco: Freeman, 1979.

5. Karp, R. M., "Reducibility Among Combinatorial Problems," Complexity of Computer Computations, Miller, R. E., and Thatcher, J. W. (eds.), 1972, pp. 95-103.

6. Lavey, R. G., "A Continuous Process Simulation Using GPSS," Proc., Winter Simulation Conference, 1981, Volume 1, pp. 347-351.

7. Rinnooy Kan, A. H. G., Machine Scheduling Problems, Martinus Nijhoff, The Hague, 1976.

8. Tsai, R., Way Kuo, K. L. McRoberts, "Using GPSS Simulation to Improve the Banbury Production System Operation," Proc. IASTED Thirteenth International Conf. Modelling and Simulation, in press (June 1985).