Proceedings of the 1991 Winter Simulation Conference
Barry L. Nelson, W. David Kelton, Gordon M. Clark (eds.)

COMBINATORIAL SCHEDULER : SIMULATION & OPTIMIZATION ALGORITHM

Toshiharu Aoki
Shusuke Nakayama
Mariko Yamamoto
Mariko Hashimoto
Jun-ichiro Tanaka

2nd Development Dept.
Production Systems Development Laboratory
NEC Corporation
484 Tsukagoshi 3-chome,Saiwai-ku,Kawasaki
Kanagawa,210 JAPAN

ABSTRACT

The recent increase of production complexity due to products mix requires a high speed, highly automated scheduling system in a factory floor. Here, we develop a new type shop scheduler based on a heuristic optimization algorithm and a discrete event simulation.

One of the features of this scheduler is high speed scheduling, which depends on the configuration of this system. The optimization algorithm determines the product sequence to be thrown into the shop. Then, a simulator determines each product flow based on the shop floor model and simple dispatching rules. The key point for high speed scheduling is to reduce the complexity of simulation rules. The simulator has responsibility only for local optimization.

A second feature of this scheduler is the scheduling algorithm which can handle multi- objectives. In the real factory, the due date has the top priority for scheduling, but the lead time and machine utilization can be changed by the production circumstances. In this scheduler, machine load leveling is used for shortening lead time and a higher load at later process is used for increasing machine utilization. An operator can assign the balance of lead time and machine utilization by setting a parameter.

An application to a real shop floor will be mentioned at the end of this paper.

1 INTRODUCTION

The major concerns in multi-item-small-batch production are to shorten production lead time, to reduce work-in-process(WIP) inventory and to improve machine utilization. The shop floor control people have to make a production schedule to meet job completion dates. Even if he has special knowledge and experience for shop floor control, the scheduling job is much too complicated and time-consuming.

To solve these problems, shop floor control people have eagerly wanted to use automatic shop schedulers. However, researches in this area, especially optimization approaches, are not successful in terms of practical use because they apply to limited conditions and domains. Recently, simulation and AI approaches for scheduling are highlighted. The focuses of these approaches are how to include the know-how of shop floor control people into their model to generate schedules with high accuracy. But, recent requirements of production, such as short lead time, less WIP, and high machine utilization, demand schedulers to include higher level optimization algorithms beyond human capability.

Based on this background, we developed a new type scheduler which generates production schedules by not only simulating the existing production line, but also optimizing the schedule using a newly introduced algorithm. The system is called SSS (triple S). The scheduling concept, system overview, and an example applied to a real production line are shown below.

2 SCHEDULING PROBLEMS

2.1 Recent Approaches

Scheduling systems, utilizing discrete-event simulation technique, or utilizing Artificial Intelligence (AI), have recently been proposed.

An advantage of the discrete-event simulation is that the production environment is easily expressed by dispatching rules. Even if the environment of scheduling is changed, the scheduler can easily be modified for the new environment by changing dispatching rules. However, the rules, in many cases, become much too complicated to generate a good schedule which meets the shop control objectives. And a lot of experiments and evaluation are required for adjusting rules with the production environment.

Many of the schedulers developed in recent years aimed to include scheduling methods done by human operators using AI technology. This AI approach is practical, but it is shop style dependent and lacks flexibility.

2.2 Problems

1)Scheduling Concept
A good scheduler must be designed based on the concept that the scheduling output should give the best material flow for a shop floor. From this point of view, it is not enough to include human scheduling methods into the system. The scheduler must have functions to automatically adjust the schedule to meet the shop management objectives by changing parameters.

2)Completeness of a scheduler
Some of the schedulers require a lot of operations such as data input/output, schedule refining, and output evaluation. These kinds of operations are time consuming for shop floor control people who have to show a production schedule within a limited time. Three key points are listed below.

- To avoid trial and errors
- To generate a good schedule automatically
- To show the results for easy evaluation

3 SYSTEM OVERVIEW

3.1 Configuration

SSS configuration is shown in Figure 1. It is composed of four blocks.

The first block is called "Load Accumulation". Usually, the daily production load from the master production schedule does not sufficiently level the production load to meet the shop capacity. This block allows an operator to check if the original plan could be finished by due date from a viewpoint of production load. Shop floor control people may change the master production schedule, or may change the shop capacity by input of overtime work. And this block determines the scheduling parameters which are used in other blocks. The parameters corresponds to the key scheduling factors, such as due date, lead time, WIP inventory, and machine utilization.

The second block is "Product Sequencing". This block determines the product sequence thrown into the shop based on key scheduling factors. The most important factor is "Due Date". Other factors are "Lead Time (WIP Inventory)", and "Machine Utilization". SSS classifies the product jobs into several groups by "Due Date", then lines up a sequence of groups evaluating the "Lead Time" and "Machine Utilization".

The third block is "Simulation". This part selects a machine and determines exact operation times for each

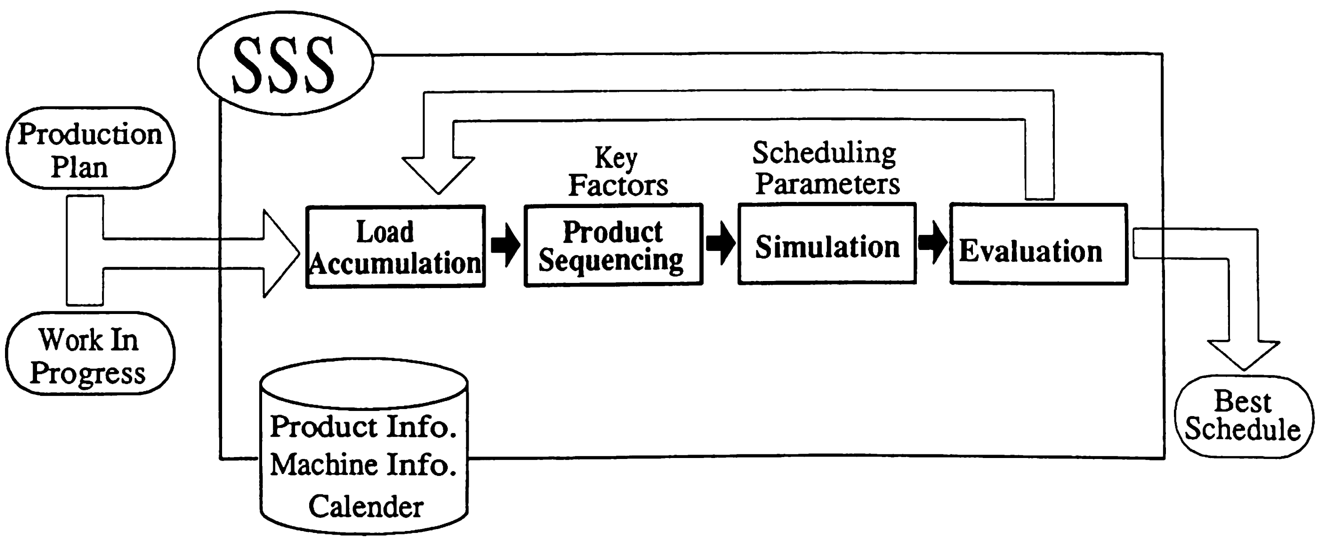


Figure 1: SSS Configuration

job checking with processing time, available machine variation and available workers, without changing the product sequence given by the "Product Sequencing" block.

The fourth block is "Evaluation". This block shows many graphs, such as machine utilization comparison, production lead time and WIP inventory transition, for an operator to use to evaluate the scheduling results. After Evaluation, if the production schedule is not satisfied, the operator changes the key scheduling factors and simulation rules. Then SSS re-schedules again for another alternative.

3.2 Two-stage Architecture

Among the four blocks of SSS mentioned above, the most important blocks are "Product sequencing" and "Simulation". This "Product sequencing" consists of heuristic optimization algorithms, and achieves a global optimization of the production schedule, considering overall production environment such as all processes and the whole scheduling period. The "Simulation", a discrete-event simulation, achieves a local optimization, considering a part of the production environment such as a single process when an event occurs. This two-stage architecture realizes a extraordinarily high speed scheduler.

Simulation may provide optimized schedules under the condition of limited machines and short time span. But, it is difficult to get effective schedules from the viewpoint of overall production environment. If overall optimization is expected for the simulation, it must have a lot of complicated dispatching rules. In this case, the processing time of the simulation becomes much too time-consuming in proportion to the number of rules, because the simulation program checks the rules every time on event occurs.

Based on the above considerations, the global optimization which does not consider detailed conditions was placed before the simulation. Then the simulation optimizes the material flow in the shop by checking processing time, state of machines and workers.

4 SCHEDULING PROCEDURE

4.1 Product Sequencing Concept

As mentioned above, the "Product sequencing" achieves a global optimization, based on the key scheduling factors. These factors are shown in Figure 2. The key factors in SSS are "Due date", "Lead time (WIP inventory)", and "Machine utilization". How each factor is handled in SSS is explained below.

1)Due date

This factor is represented as the remaining slack (= due date - processing time - current date) for each job. That is, the jobs which have less slack get higher priority. In the program, jobs are sorted in increasing order of slack, then other key factors are considered within an allowable range.

2)Lead time (WIP inventory)

This factor accounts for the machine load in a shop. Jobs will be sequenced so that machine load in all processes will be leveled. Machine load leveling aims at lower WIP inventory and shorter production lead time.

3)Machine utilization

This factor accounts for products grouping and a reasonable WIP inventory.

The same products or similar products are grouped in the product sequence. However, to decrease setup time at all processes, not only the "Product Sequencing" block, but the "Simulation" block has to handle this grouping concept.

Increase in WIP inventory up to a reasonable level reduces machine idle time. In particular, the bottleneck machine should always have several waiting jobs to achieve high machine utilization.

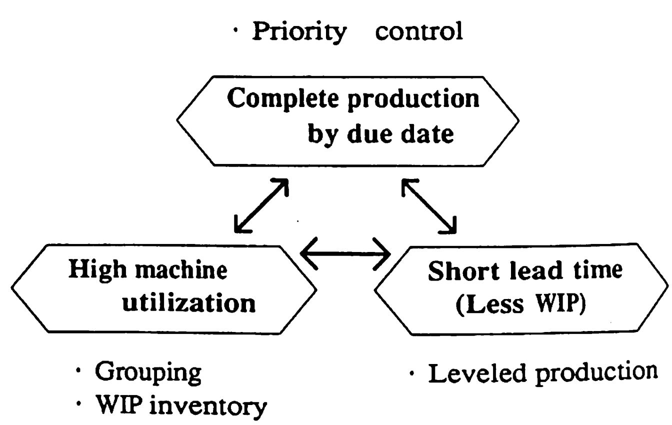


Figure 2: Key Factors

4.2 Product Sequencing Algorithm

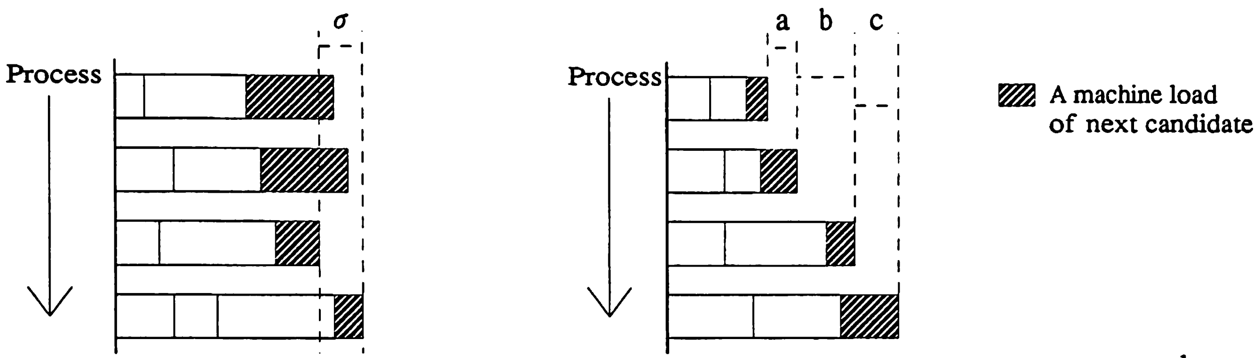
Under this concept, the program generates the product sequence. Here, the balance adjustment between "Lead Time" and "Machine Utilization" is focussed. Evaluation equation(1) is shown in Figure 3. The way to get the lead time evaluation value differs from that of machine utilization evaluation value. The details will be explained below. In equation(1), the ratio α is a weight coefficient of lead time. The ratio 1-α is of machine utilization. The coefficient α is given by an operator prior to scheduling. In SSS, a job whose evaluation value is the smallest, is selected as the next job.

1)Lead time evaluation value (Figure 3 (a))

First, machine loads for all jobs whose sequence have already been determined are accumulated at each process. Then, a machine load of the next candidate job is accumulated to the pre-calculated loads at each process. At this point, the variability of these loads, shown as the standard deviation σ in Figure 3 (a) , is the evaluation value.

$$\text{Evaluation value} = \alpha \cdot \text{Lead time evaluation value(a)} + (1-\alpha)\cdot \text{Machine utilization evaluation value(b)} \quad (1)$$

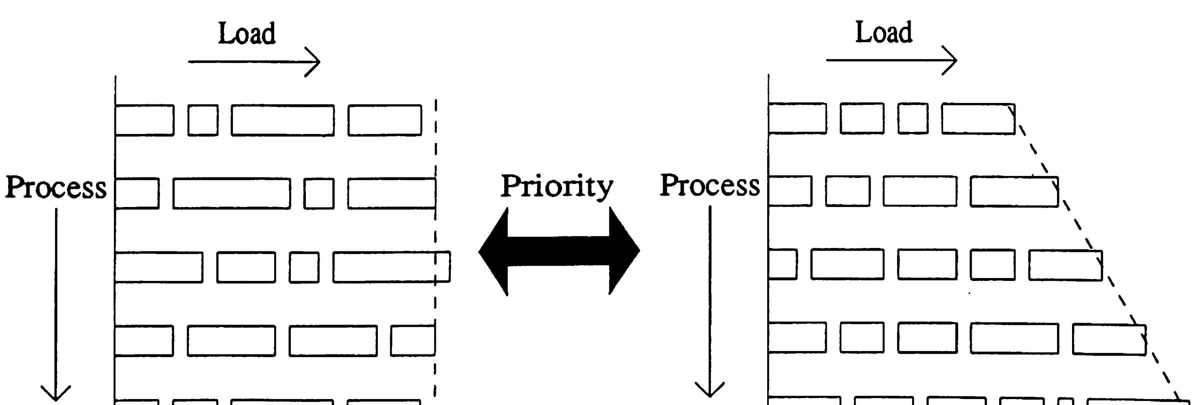
$$\alpha = \text{weight } (0 \leq \alpha \leq 1)$$



(a) Lead time evaluation value = σ Standard deviation of loads at each process

(b) Machine utilization evaluation value = $\frac{1}{a+b+c}$ Reciprocal sum of loads difference between a process and the next one

Figure 3: Evaluation Equation



(a) Higher Priority To Short Lead Time

(b) Higher Priority To Machine Utilization

Figure 4: Evaluation Value

2)Machine utilization evaluation value (Figure 3 (b))

Similar to the lead time evaluation, the candidate's load is accumulated to previous loads at each process. At this time, the reciprocal sum of the load difference between a process and the next one (a,b,c in Figure 3 (b), both plus and minus value), is the evaluation value.

The result of machine load accumulation is shown in Figure 4. The way to accumulate the loads is changed from (a)(higher priority to short lead time) to (b)(higher priority to machine utilization) by adjusting the ratio α. When the ratio α is close to 1, the accumulation is leveled, and lead time becomes short. When the ratio α is close to 0, that is, the accumulation of the first process loads is much less than the later process loads, the machine utilization becomes high.

4.3 Simulation Concept

The "simulation" has responsibility for local optimization of the production schedule. Key points of the simulation to generate a good schedule and to be used in a shop floor are shown below.

1)Group technology

Group Technology (GT), which is used in dispatching rules, helps scheduling people to incorporate the shop floor operating rules into the model. Job grouping structure, which may be categorized by GT code, is hierarchical. An example of such a hierarchy is shown in Figure 5. Jobs of the first layer are grouped by due date or priority. The order exists among these groups. Next, the second layer jobs are grouped by product types. Last, the third layer is differentiated by each job.

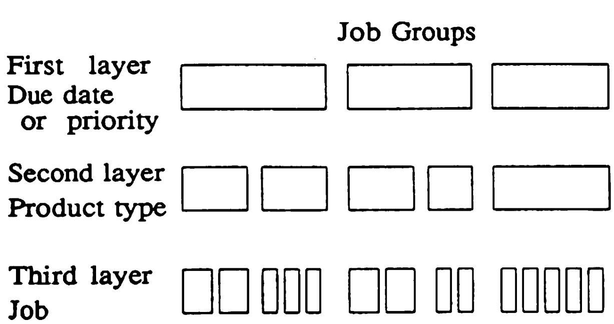


Figure 5: Example of Hierarchy Grouping

2)Local optimization

The material flow of the simulation is determined by the dispatching rules. A suitable combination of dispatching rules can generate an optimum schedule for the process in which the dispatching rules are defined.

Dispatching rules are written based on the consideration of product sequence, restriction, due date, lead time and machine utilization. Furthermore it is possible to have a hierarchical structure for combinations of dispatching rules using IF-THEN format.

The details of the dispatching rules are explained in the next section.

3)Relationship to previous schedule

The previous scheduling result should be considered in the current scheduling. A part of the current schedule must succeed to the previous machine assignment to avoid human confusion and additional setup of machines.

4.4 Simulation Dispatching Rules

Dispatching rules are the key for material flow as mentioned above. The rules are shown in Figure 6.

1)The first process

Dispatching rules of the first process are shown in Figure 6 (a). The operation sequence of this stage should be basically equal to the production sequence given by the "Product sequencing" block. Because of this, the rules are FIFO (First In First Out).

Furthermore, WIP inventory in a shop should not rise beyond a reasonable inventory level. The first process does not start processing jobs in SSS when WIP inventory level is over the preset number. At the time WIP inventory becomes lower, processing starts again.

2)Other processes

Dispatching rules for other processes are shown in Figure 6 (b). A job which has the same GT code as the current job has is sent to the next operating sequence. When there is no job which has the same GT code, a job which has a similar GT code is selected. But, if a job which does not have enough remaining slack exists, this job may get the highest priority. The combination of different rules can be written using IF-THEN format.

```
(a)The First Process

  PROCESS : SMT
     IF priority > 5
         THEN
            FIFO
         ELSE IF
               next process inventory < 10
            THEN
               FIFO
  END-RULE


(b)Other Process

  PROCESS : INSERTION
     IF remaining slack > 20H
         THEN
            CHOOSE SIMILAR GT CODE
            CHOOSE SAME GT CODE
         ELSE
            PRIORITY ORDER
  END-RULE
```

Figure 6: Dispatching Rules

5 EVALUATION

The "Evaluation" block helps the operator to evaluate the schedule generated by the "Product sequencing" and the "Simulation" blocks. Gantt charts are widely used for presentation of scheduling results. But, it is difficult to determine if the schedule is satisfactory for the objectives. The evaluation block of SSS calculates such production measures as machine utilization, lead time and due date satisfaction.

The outputs are shown in Figure 7, Figure 8 and Figure 9. Figure 7 shows a gantt chart that indicates the job sequence at each machine. Figure 8 shows a gantt chart that indicates the operation sequence for each job. Figure 9 (a) shows a radar chart that indicates balance of three requirements, due date satisfaction, lead time and machine utilization. Figure 9 (b) shows the machine utilization that indicates the ratio of utilization and setup time. Figure 9 (c) shows an output chart that indicates the number of finished goods for each hour and cumulatively.

Shop control people check these graphs from various points of view to see if the schedule is satisfactory. If they are not satisfied with the results, they can change the parameters of the "Product sequencing" and dispatching rules of the "Simulation". After that, SSS schedules again.

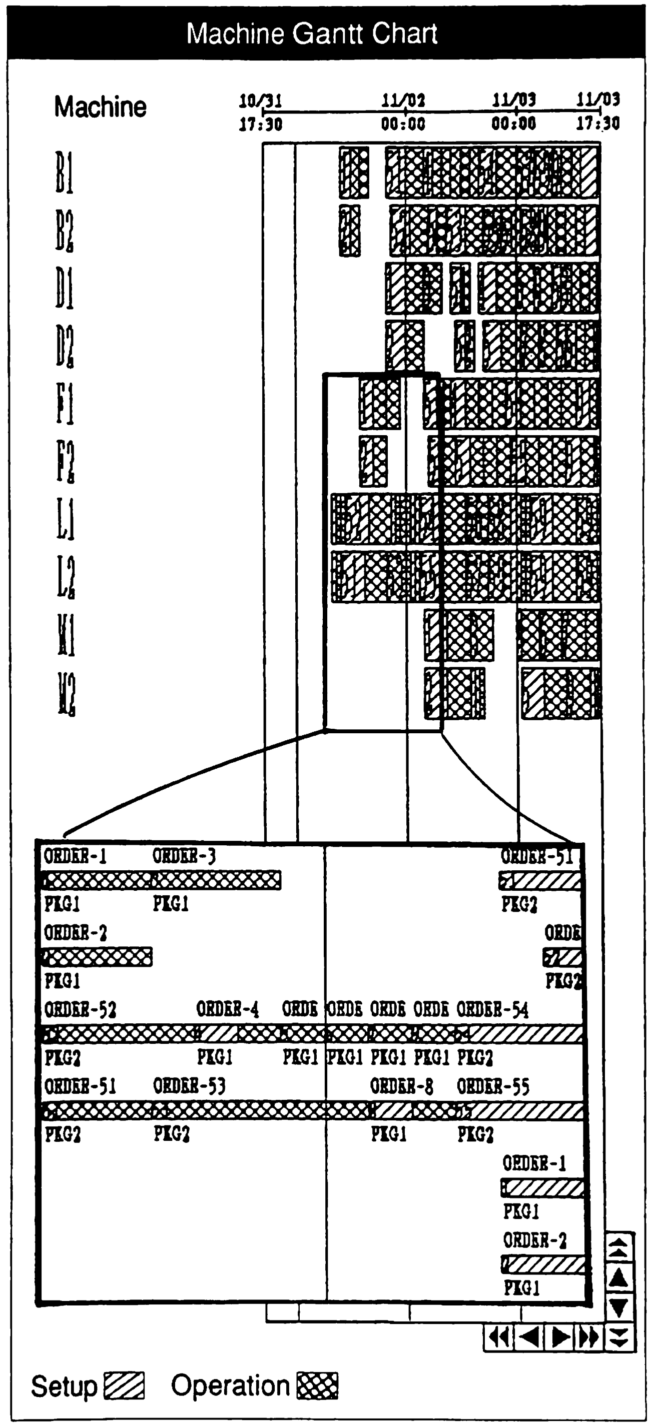


Figure 7: Machine Gantt Chart

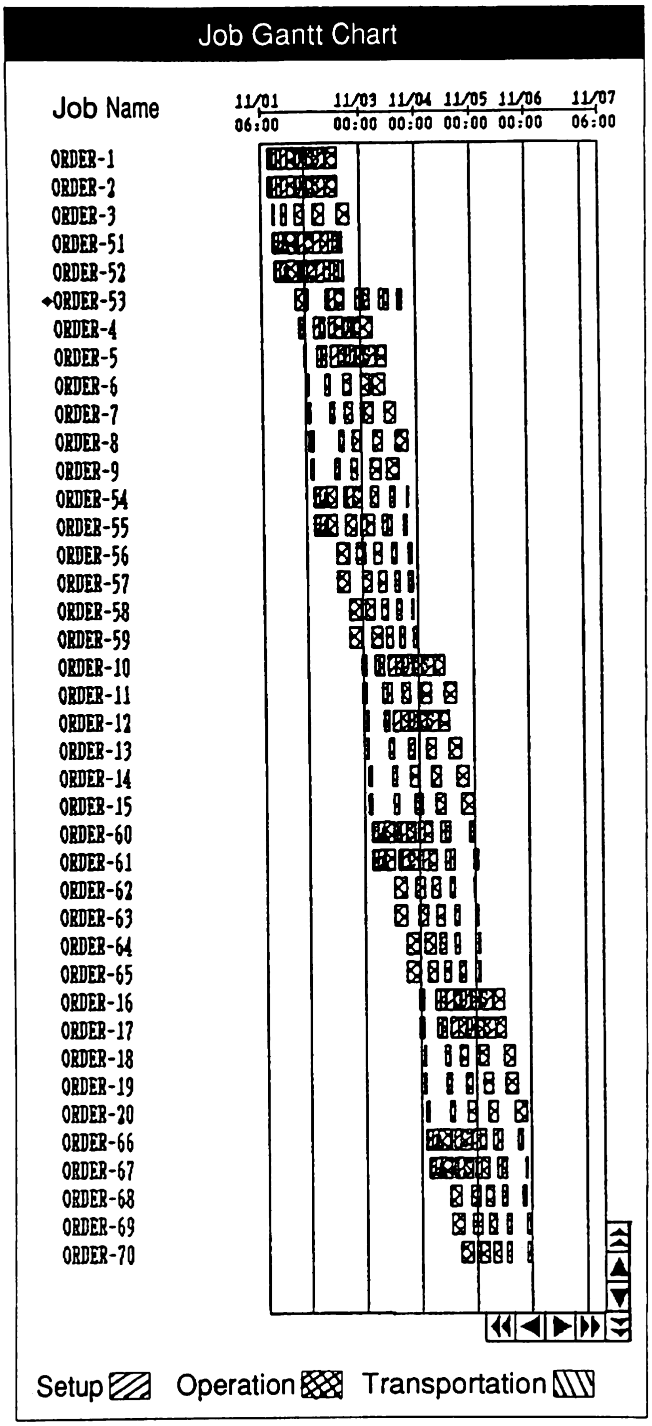
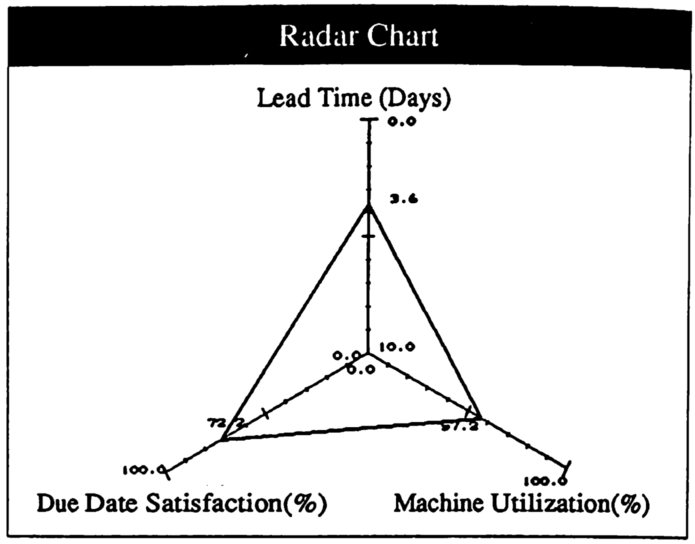
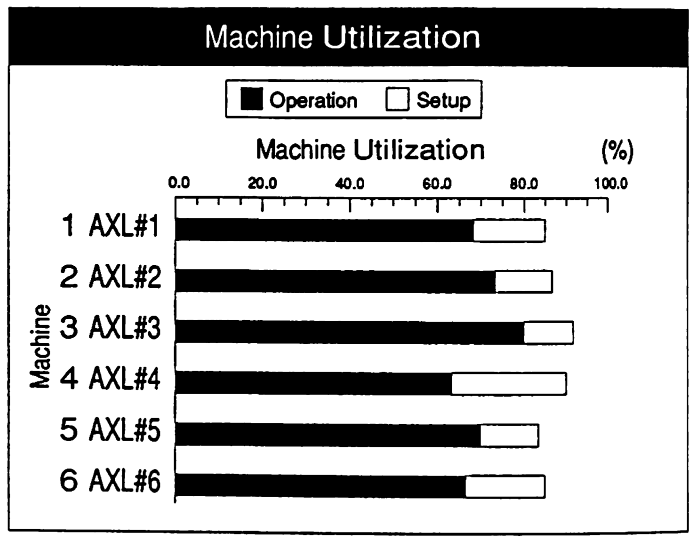


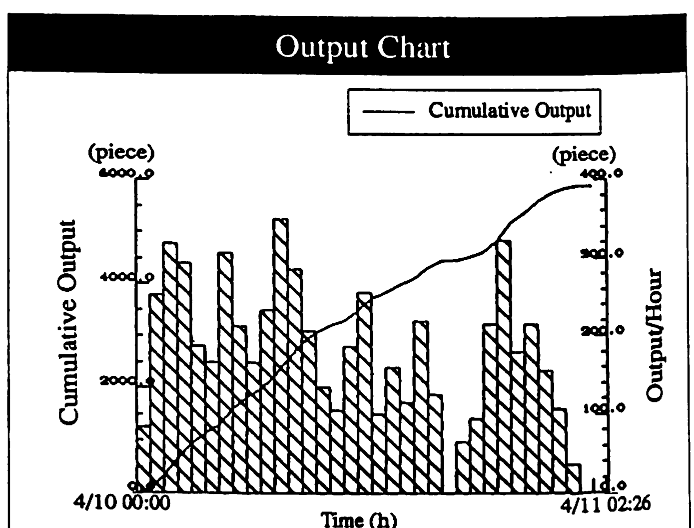
Figure 8: Job Gantt Chart



(a) Radar Chart



(b) Machine Utilization



(c) Output Chart

Figure 9: Evaluation Output

6 CAPACITY AND SCHEDULING TIME

Capacity and scheduling time of this scheduler are shown in Table 1.

Table 1: Capacity And Scheduling Time

(a) Capacity

Shop Type	:	Flow Shop , Job Shop
Job MAX.	:	1000
		Job × Process = 10000
Product MAX.	:	1000

(b) Scheduling Time

Sucheduling Time 2 Minutes
(Under the Condition Number of Jobs : 300 Number of Processes : 6)

This schedule is run on an NEC engineering work station (UNIX).

7 APPLICATION

This scheduler has been used for daily scheduling in a PCB assembly shop at NEC for 1year. There are 200 kinds of products and 300 jobs in the shop. Each product has a different route through a series of processing operations. There are six processes for each product on average, ten processes at a maximum. Each process has 2 to 3 machines.

The time required for scheduling is about 20 times faster than that of manual scheduling. As a result, the shop lead time became half, and the setup time was reduced to one tenth, of their previous values.

8 CONCLUSION

This paper describes requirements of production, required functions for schedulers and a proposed scheduling system.

The SSS, a scheduler with four blocks, generates a schedule that satisfies multiple requirements, due date, lead time and machine utilization. The two-stage architecture adopted in this system greatly improved the scheduling speed in comparison with schedulers based on AI or simulation. This system certified that two-stage architecture, "Product sequence" and "Simulation", is useful.

However, there are several ways in which the system could be improved.

1)How to handle feedback?

How to change the parameters and rules, if a scheduling result is not satisfactory?

2)Scheduling speed

The time, to make schedules, became short. But, pre-process (data download and load to memory) and after process (data upload) takes a while (5 - 15min).

3)Variety

The "Simulation" model can be easily changed for other shops by changing dispatching rules. But, the "product sequencing" model has limitations because of its algorithm.

REFERENCES

Frederick A.Romammer and K. Preston White,Jr. A Recent Survey of Production Scheduling. *IEEE Transactions on systems, man,and cybernetics Vol. 18, No. 6, November/December 1988* 841-851

Masayuki Enomoto, Mariko Yamamoto and Shusuke Nakayama. A Scheduling System With Knowledge-Base and Simulation. In Proceedings of *the 1990 International Conference on Manufacturing Systems and Environment*, 437-441.

AUTHOR BIOGRAPHIES

TOSHIHARU AOKI received his Bachelor of Science and Engineering degree in systems engineering from Tokyo Denki University in 1986. He joined NEC Corporation in 1986, and has been engaged in the research and development of computer application systems in the field of production scheduling and control systems. He is now member of the 2nd Development Department, Production systems Development Laboratory.

SYUSUKE NAKAYAMA received his B.E. degree in mechanical engineering from the University of Tokyo in 1977, and his M.S. in industrial engineering from Purdue University in 1985. He joined NEC Corporation in 1977, and has been engaged in the research and development of production automation and control systems. He is now Engineering Manager of the 2nd development Department, Production Systems

Development Laboratory. He is a member of the Institute of Industrial Engineering.

MARIKO YAMAMOTO received her B.S degree in mathematics from Josai University in 1982. She joined NEC Corporation in 1982, and has been engaged in the research and development of simulation and its application software. She is now Supervisor of 2nd development Department, Production Systems Development Laboratory.

MARIKO HASHIMOTO received her B.E. degree in electrical engineering from Tokai University in 1988. She joined NEC Corporation in 1988, and has been engaged in the research and development of computer application systems in the field of production scheduling. She is now a member of the 2nd Development Department, Production Systems Development Laboratory. She is a member of the Information Processing Society of Japan.

JUN-ICHIRO TANAKA received his B.E. degree in electronics-communication engineering from Meiji University in 1989. He joined NEC Corporation in 1989, and has been engaged in the development of computer application systems in the field of artificial intelligence. He is now a member of the 2nd Development Department, Production Systems Development Laboratory.