# A Study of a Scheduling Simulator for the Analysis of Multi-stage Flowshop Scheduling Problem with eM-Plant

**Y. Futatsuishi** [a], **I. Watanabe** [b] **and T. Nakanishi** [c]

[a] *Faculty of Engineering, Seikei University, Tokyo, 180-8633, Japan (futatsu@is.seikei.ac.jp)*

[b] *Faculty of Engineering, Seikei University, Tokyo, 180-8633, Japan (watanabe@is.seikei.ac.jp)*

[c] *Faculty of Policy Studies, Tokushima-bunri University, Tokushima, 770-8514, Japan (tosio@blue.ocn.ne.jp)*

**Abstract:** In small-item large lot production, one work person only had to do single processing. A recent tendency of the production system is the multi-item small lot production. In this type of production system, one work person should have a lot of skills adaptable to multi-item articles. Because of the fact that one work person has to have more than one skill, one processing operation which composes a job has come to be handled by several work persons. From the viewpoint of a job, one elementary operation has come to be processed by several alternative operation assignments. The scheduling problem mentioned below assumes the operation under the above-mentioned situation. The n jobs processed by three work persons (or machines) are given, and either of n jobs is comprised of 5 elementary operations. As for the 1st, 3rd and 5th operations, they are assigned to certain work persons. As for the 2nd operation, it can be processed by either the work person who processes the 1st or the work person who processes the 3rd operation. As for the 4th operation, it can be processed by either the work person who processes the 3rd or the work person who processes the 5th operation. Under these conditions we discussed the problem of how to minimize the total elapsed time from the starting time up to the completion time of a job. At present, systems which have scheduling functions as a part of production management software have been developed. The theoretical analysis of scheduling problems is difficult and we have to rely upon the approximation method or simulation methodology. This paper aims at developing a scheduling simulator for the theoretical analysis with object-oriented simulation language eM-Plant (Old product name is Simple++).

***Keywords:*** Simulation; Scheduling; Assignment; Flowshop

## 1. INTRODUCTION

For the theoretical analysis of the scheduling problem, it is very important to know the characteristics of the optimal solution based on the example of the numerical value. Moreover, it is necessary to analyze the data obtained by a number of numerical simulations to compare and to evaluate two or more approximation methods. It is also necessary to prepare the program respectively for the production system with various structures to analyze the above-mentioned simulation results. Therefore, faculty students' load of the programming was heavy.

Wada et al. [1997] clarified input and output information needed to design an analytical system of this problem. This analytical system has been developed with eM-Plant which is one of the simulation languages. In addition, Wada et al. [1998] gave the hierarchy of the structures of the model and clarified necessary information according to the hierarchy. However, it was still difficult to know the characteristics of the optimal solution and to compare and to evaluate two or more approximation methods.

Thus, this study aims at developing the simulator which enables the following two properties:

- The model hereof can be built without complex programming techniques.
- Developing and modifying the approximation method are facilitated.

## 2. DEVELOPMENT POLICY OF THE SYSTEM

This chapter describes the outline of the scheduling model which has been treated in the laboratory. The development policy and the simulation language used to analyze our model are explained.

### 2.1 Outline of the Model Upper and Lower case

A variety of scheduling problems have been taken up in our laboratory. One among them is the problem of operation assignment. For instance, there is a machine that can process many kinds of operations. In this case, one operation can be processed by two or more machines. From the viewpoint of processed items, one elementary operation can be processed by several alternative operation assignments. To decide one schedule when the assignment of processes to machines are flexible, two necessary decisions are: "The assignment of each elementary operation has to be requested" and "The processing sequence in each process has to be decided". When the assignment of processes to machines is fixed, the schedule can be made by deciding the processing sequence [Johnson, 1954]. Therefore, the problem taken up here proved to be more difficult.

Figure 1 shows the model of three-stage flowshop scheduling problem with alternative operation assignments treated by this study.

Each processed item i(i =1,2,...,n) has five elementary operations <ai>, <bi>, <ci>, <pi> and <qi>. Elementary operations <pi> can be processed by either M1 or M2. Elementary operations <qi> can be processed by either M2 or M3. Under these conditions we discussed the problem of how to minimize the total elapsed time between the starting and completion time of a processed item.

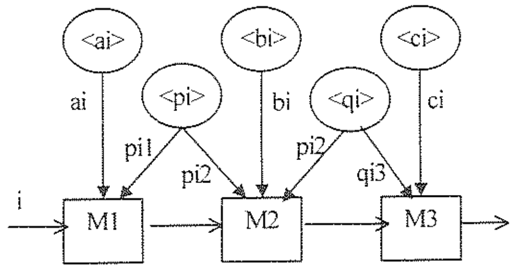


**Figure 1.** The model dealt with in our study.

### 2.2 Formulation of the Problem

Formulation of the problem consists of the following 7 steps:

- The n processed items are given and each processed item goes through the processes of M1, M2 and M3 in this order.
- Each processed item i(i =1,2,...,n) has five elementary operations <ai>, <bi>, <ci>, <pi> and <qi>. <ai>, <bi> and <ci> are processed by M1, M2 and M3 respectively. <pi> can be processed by either M1 or M2. <qi> can be processed by either M2 or M3.
- Processed item i is called (I,II)-type job when <pi> is assigned to M1 and <qi> is assigned to M2, (I,III)-type job when <pi> is assigned to M1 and <qi> is assigned to M3, (II,II)-type job when <pi> is assigned to M2 and <qi> is assigned to M2, and (II,III)-type job when <pi> is assigned to M2 and <qi> is assigned to M3.
- Processing time of production elements <ai>, <bi>, <ci>, <pi> and <qi> for each processed item i is given by ai, bi, ci, pij and qij respectively. pij designates the processing time of <pi> on Mj (j=1,2). qij designates the processing time of <qi> on Mk (k=2,3).
- Only one job can be in process on one operation at a time. Once an operation starts on Mj (j=1,2,3), another operation has to wait until the preceding operation is over.
- Mj (j=1,2,3) can handle at most one operation at a time.
- Each processed item is available at time zero.

### 2.3 Development Policy

By now, Futatsuishi et al. [1999] have proposed the approximation method on this problem. It is necessary to understand the feature of the problem to understand the approximation method in depth. The program has been written by C language, through which we executed numerical simulation. In that case, the following problems occur.

- It takes time to work on the programming and it is not possible to reach a theoretical conclusion easily.
- The completed program takes into consideration only the structure of the target scheduling. Therefore, it lacks generality.
- The execution result of the program and its movement are not visible.

The following three points are set to development policy.

- Not only the model but also the method are processed like parts (Parts are constituent parts of a system). The model and the method can be easily constructed only through the arrangement procedures.
- The structure, which leads to easy model building and modification, is attained.
- The simulation results and the model behavior made are visualized.

### 2.4 Language Development

Recently, package software for efficient simulation has been developed by the advancement of computer technology. eM-Plant is an object-oriented simulation language, and a model can be built comparatively easily.

A basic concept of eM-Plant is to be able to model the production line by using 38 basic construction parts prepared by the standard. These basic construction parts can be treated as objects. As a result, eM-Plant has the following two features:

- Basic construction parts are arbitrarily arranged on the Frame object which is one basic construction part. Also, the user-defined object can be made by editing them. This object is registered as a basic construction part of eM-Plant, and the same object can be reproduced. The reproduced object can be changed at the same time by using the succession concept.
- An object in the upper layer can be made by arranging a basic construction part and the user-defined objects on the frame object as a new object.

Thus, the change and the expansion of the system can be easily done by using hierarchical concept and succession concept of eM-Plant. This system has been developed by eM-Plant of Windows application.

## 3. SIMULATION SYSTEM AND PROBLEM SOLVING

The content of the system is explained by the following Sections.

### 3.1 Necessary Input and Output Information for the System

Necessary input and output information for this system is enumerated as follows based on the assumed conditions in Section 2.2.

Necessary input information for the system

- Number of job
- Processing sequence
- Processing time
- Due date
- Dispatching rule

Necessary output information for the system

(a) Information on the job

- Arrival time to queue
- Waiting time to queue
- Waiting time to machine
- Lead time
- Flow time
- Total waiting time
- Lateness
- Tardiness
- Earliness
- Completion Time

(b) Information on the entire job

- Max lead time
- Max Lateness
- Max tardiness
- Max Earliness
- Max total waiting time
- Max flow time

(c) Information on the entire schedule

- Completion time
- Total flow time
- Average flow time
- Total lead time
- Average lead time
- Total lateness
- Average lateness
- Total tardiness
- Average tardiness
- Total earliness
- Average earliness
- Tardiness job

(d) Information on working of machine

- Working time
- Idle time

The following Section 3.2 explains the generation process of the solution through objects-oriented and structured programming.

### 3.2 Generation Process of Solution

The generation process of the solution is as follows:
(1) The job with the processing time is generated.
(2) The processing order is decided.
(3) The assignments of the elementary processes are decided.
(4)The solution is obtained according to the model.
(5) The solution is evaluated according to the end condition.
Returns to (2) when a better solution is needed after evaluation. Advances to (6) when ending.
(6) The job is terminated.
However, the order of (2) and (3) can be different depending on the approximation method.

### 3.3 Generation of Objectives

The items shown in the foregoing Section 3.2 is made an object as follows:

- START MODEL is an object which generates the job with the processing time. (Section 3.2 (1))
- SEQUENCE MODEL is an object which decides the processing order. (Section 3.2 (2))
- JOBTYPE MODEL is an object which decides the allocation of the elementary work. (Section 3.2 (3))
- ASSIGNMENT MODEL is an object which represents the model. The model of Figure 1 is shown at this study. (Section 3.2 (4))
- CONDITION MODEL is an object which judges the end condition of the approximation method. (Section 3.2 (5))
- END MODEL is an object which deletes the job. (Section 3.2 (6))

The display screen of the monitor is shown in Figure 2 as an example of (2) - (5) in Section 3.2.

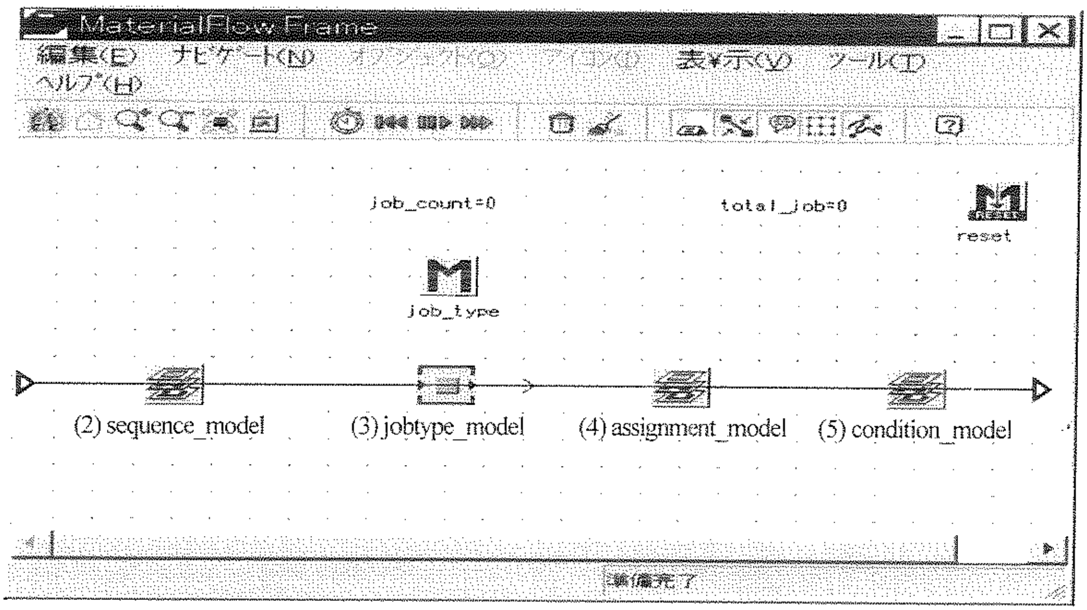


**Figure 2.** Object making with eM-Plant.

This part is different depending on the model and the approximation method of the object. For instance, the method of deciding the assignment can be changed to another approximation method by replacing the part of (3) in Figure 2 with other JOBTYPE MODEL objects

### 3.4 Structure of Model

The generation process of the solution shown by Section 3.2 is expressed by the layered structure shown in Figure 3. The role of each hierarchy of Figure 3 is as follows.

- It is the most significant hierarchy of the model (Figure 4). This object (Setting of job in Figure 3) can control the start and the end of the simulation.

The model can be made by hierarchically arranging the following object.

- It is an object (Generation of job in Figure 3) which gives the attribute values like the processing time, numbers of jobs, etc. and generates the job. The object in foregoing Section 3.3 (START MODEL) is arranged.
- It is an object (Flow of job in Figure 3) by which information on the flow of the job of the processing order, the type of job, etc. are decided. The object of foregoing Section 3.3 (SEQUENCE MODEL, JOBTYPE MODEL, ASSIGNMENT MODEL and CONDITION MODEL) is arranged.

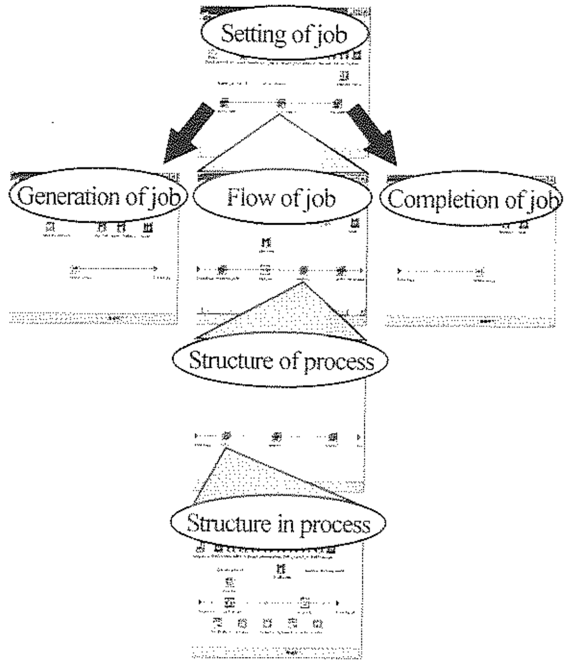


**Figure 3.** Layered structure of modelling.

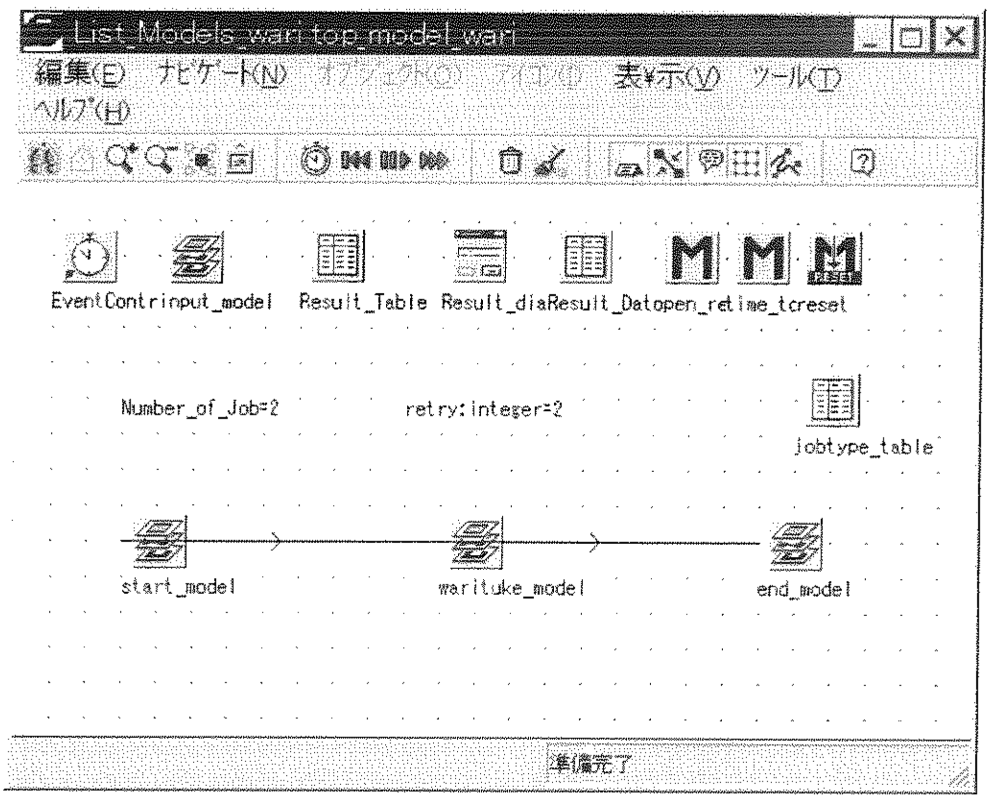


**Figure 4.** Object which sets the job.

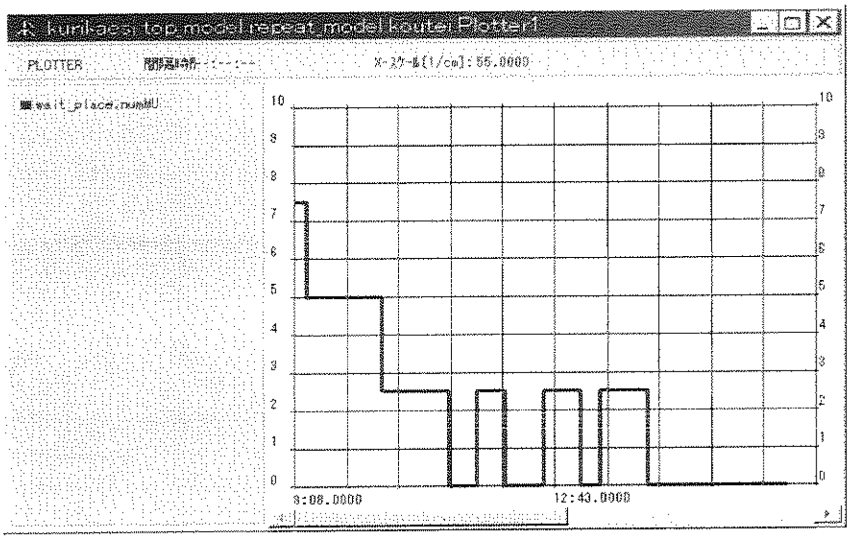


**Figure 5.** Waiting number of job in queue.

- The waiting time of each process, the processing start time and the end time of the job which terminates all processing are recorded. It is an object (Completion of job in Figure 3) which deletes the job on the model. The object in foregoing Section 3.3 (END MODEL) is arranged.
- It is an object (Structure of Process in Figure 3) which builds the model. The object of foregoing Section 3.3 (ASSIGNMENT MODEL) is arranged. The approximation methods can be compared.
- It is a part where actual processing is done. It is an object (Structure in Process for Figure 3) which shows the queue composed of jobs and the middle product depository for a processed machine.

## 4. CONCLUSIONS

The conclusion of the system shown by this study is as follows:

- The model of the problem containing the operation assignment has come to be built in a short time.
- The object-oriented modelling of the solution generation process has made the overall image of solution method easy to grasp.
- It has become easy to develop the solution method and to modify the model.
- The model behavior can be observed visually (Figure 5), which has made the evaluation analysis of the solution easier.
- The comparison of the various approximation methods has become easier.

The following is a future problem. The procedure of the approximation method should be subdivided further. Generality and extensibility of the approximation method should be improved by object-oriented methodology.

## 5. REFERENCES


Wada.H, Y.Futatsuishi, and I.Watanabe, The Development of a scheduling simulator for the analysis of scheduling problem with Simple++, The 16th Japan Society for Simulation Technology Conference, pp.283-286, 1997.

Wada.H, Y.Futatsuishi and I.Watanabe, The Proposal of a scheduling simulator for the analysis of scheduling problem with Simple++, The 17th Japan Society for Simulation Technology Conference, pp.283-286,1998.

Johnson,S.M., Optimal two-and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly* in English, Vol.1,pp.61-68,1954.

Futatsuishi.Y, I.Watanabe and T.Nakanishi, A study of multi-stage flowshop scheduling problem with alternative operation assignments, In: Proceedings of The International Congress on Modeling and Simulation, Hamilton, New Zealand, vol.4, pp923-928, 1999.