# KNOWLEDGE-BASED SCHEDULING FOR FLEXIBLE MANUFACTURING SYSTEMS

Ying C. Yang, Ming C. Liu and Jeffery K. Cochran

Department of Industrial and Management Systems Engineering
Arizona State University, Tempe, Arizona 85287

## ABSTRACT

Research in the field of scheduling problems has applied the techniques of artificial intelligence (AI) to real-time scheduling in job shop production. In this paper, an expert system is developed to solve the scheduling problems for Flexible Manufacturing Systems (FMS). The FMS environment is described by the semantic network which contains the manufacturing organization information as a knowledge base. A Backward Constraint Search technique, embedded in the inference engine of the expert systems (ES), is used to extract information from the knowledge base and generate the final schedule. When there is no feasible solution or there are number of conflicting solutions, pairwise comparison of Analytic Hierarchy Process (AHP) is employed as the decision tool to solve the scheduling problems. This system is written in Turbo Prolog and is designed to implement scheduling on microcomputers at the shop-floor level.

## INTRODUCTION

Flexible manufacturing systems are automated batch manufacturing systems which produce parts with the efficiency of mass production systems, and the flexibility of job shops -- all under integrated computer control (Kiran and Tansel, 1985). FMS is characterized by limited resources, frequently changed tools, and the sequence of job operations. Based on these characteristics, the FMS scheduling should reflect the manufacturing environment as well as the real-time control issues. The importance of scheduling in FMS results from the high capital investments involved. Trying to reduce the production cost in FMS by solving the scheduling problem is more difficult than in other production systems, because of the complexity of the manufacturing environment.

The heuristic rules approach to finding a nearly optimal scheduling, based on some assumptions, has been developed for job shop scheduling problems (Hershauer and Ebert, 1974; Dar-El and Wysk, 1982; Kiran and Smith, 1984; and Bunnag and Smith, 1985). A job shop scheduling problem is an NP complete problem: that is, the number of potential solutions increases exponentially with system complexity. Although heuristic rules are the most common approach in practice, heuristic rules have not yet defined a general approach for FMS scheduling, as indicated by King and Spachis (1980).

Artificial intelligence (AI) techniques have an application to scheduling problems. The first successful AI system for job shop scheduling was developed by Mark Fox at Carnegie-Mellon University in 1983. Steffen and Greene (1986), and Wu (1987) also employed AI techniques to solve scheduling problems. Shaw and Whinston (1986) developed an expert system for scheduling in FMS which uses nonhierarchical planning to perform the solution searching. Morton and Smunt (1986) developed a generic system, PATRIARCH, which integrates hierarchical structure, decision support capability, advanced knowledge representation, and accurate large scale "shadow price" heuristics. PATRIARCH is the first system that combines operations research (OR), decision making (DM) and AI for FMS scheduling problems. These knowledge-based approaches provide guidelines for an alternative way to solve scheduling problems in FMS.

Characteristics of FMS that affect decisions are the limited resources available, business polices, and production control strategy. Saaty (1980) developed AHP to enable effective decisions to be made on complex issues by simplifying and expediting the natural decision-making processes. A general schema to prevent inconsistency in comparison and to solve conflicting solutions is proposed by Adiga and Cochran (1985) with an expert system using AHP analysis. In this paper, a new OR/AI/DM approach for FMS scheduling is developed, and the results of its implementation are presented.

## MODEL STRUCTURE

The structure of an expert system (ES) generally contains three components: knowledge database, inference engine, and control. The collection of rules which represents the expert's knowledge is called a knowledge database, where semantic networks are used as the knowledge representation of an ES/FMS scheduling. An inference engine is a program which uses the knowledge database to produce an expert recommendation with complete and incomplete information. The most common structure of the inference engine is a series of "if-then" statements. Control is the reasoning deduction performance of an inference engine. Two approaches, forward chaining and backward chaining, are employed by the inference engine for reasoning deduction.

When more than one operation is ready for processing at a particular station, a specific dispatching rule must be applied. The most common dispatching rules in scheduling for job shop are: SPT (shortest process time), MWKR (most work remaining), LWKR (least work remaining), and MOPNR (most operation remaining). The selection of dispatching rules should be based on the manufacturing environment. The dispatching rule's effectiveness is measured by the value of the criterion function. The measuring criteria for job shop performance of dispatching rules used in this paper are: job due date, machine utilization, makespan, and work-in-process inventory cost.

In the AHP analysis, the scale to the criteria comparison result is based on both the scheduler's experience and the company's policy; the comparison of goals is based on the information calculated from the inference engine. The scale assigned to each rule depends upon its degree of importance under a certain criterion. Once the scales are assigned to the comparison of all criteria, AHP analysis determines the best schedule based on a specific dispatching rule for the current environment. If no feasible solution occurs, the relaxation of due date constraints can also use AHP analysis. Three criteria considered in due date relaxation are: penalty/profit loss, company's reputation loss, and customer satisfaction. Through the AHP analysis which job's due date should be relaxed can be determined.

The structure of the semantic network for an FMS environment is shown in Figure 1:

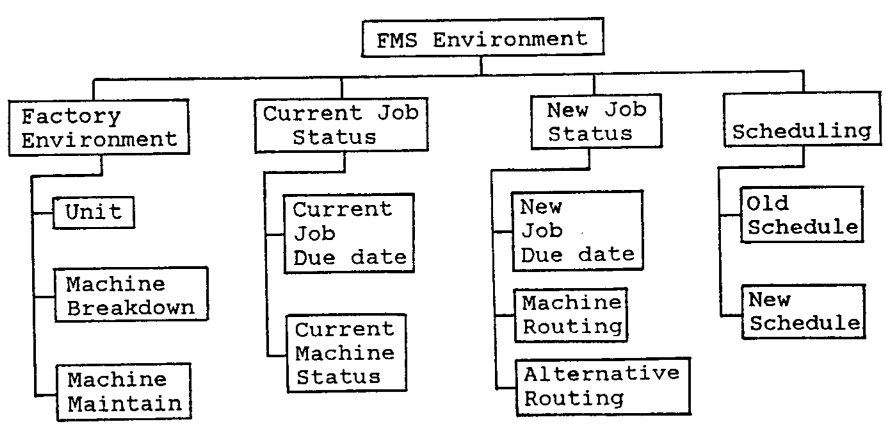


Figure 1: Semantic Networks Representing an FMS Environment

This model has five key assumptions:

(1) Pre-emption is allowed only for rescheduling.
(2) Due dates are fixed.
(3) Each job operation has alternatives to be machined on different machines.
(4) Different job operations can be machined on the same machine.
(5) Process time, setup time, and transportation time are deterministic.

This model uses the following notations:

Di : the due date of job i.
SSt : the search space at state t.
Jij : the jth operation of job i.
PTijkt : the process time of Tij on machine k at state t.
Mk : the machine k.
CTijkt : the time at which Tij is completed on machine k at state t.
STijkt : the starting time at which Tij can be machined on machine k at state t.
MTkt : the time at which machine k can be assigned to complete an operation during backward constraint search at state t.
BETk : the time at which machine k completes maintenance, or the end of break down.
BSTk : the time at which machine k starts maintenance or breaks down.

The term "state" in the notation is defined as the situation in which only one of the schedulable operations can be scheduled. Moreover, the "search space" is the set of schedulable operations at the specific state. Each schedulable job/operation is a node in the search space, and its data structure is

| | | | | |
|---|---|---|---|---|
| Jij | Mk | PTijkt | CTijkt | STijkt |

The following section discusses the methodology of finding the near-optimal knowledge-based FMS scheduling.

## METHODOLOGY

The overall methodology structure is shown in Figure 2. Once the knowledge representation of an FMS environment is developed, the inference engine conducts four dispatching rules to generate, at most, four near-optimal schedules. The AHP system determines the best schedule, even though there is no feasible solution generated by the inference engine. The knowledge representation structure is based on the hierarchical planning which decomposes an FMS environment into four components: factory environment, current job status, new job status, and scheduling. Each component is described by several predicates, as follows:

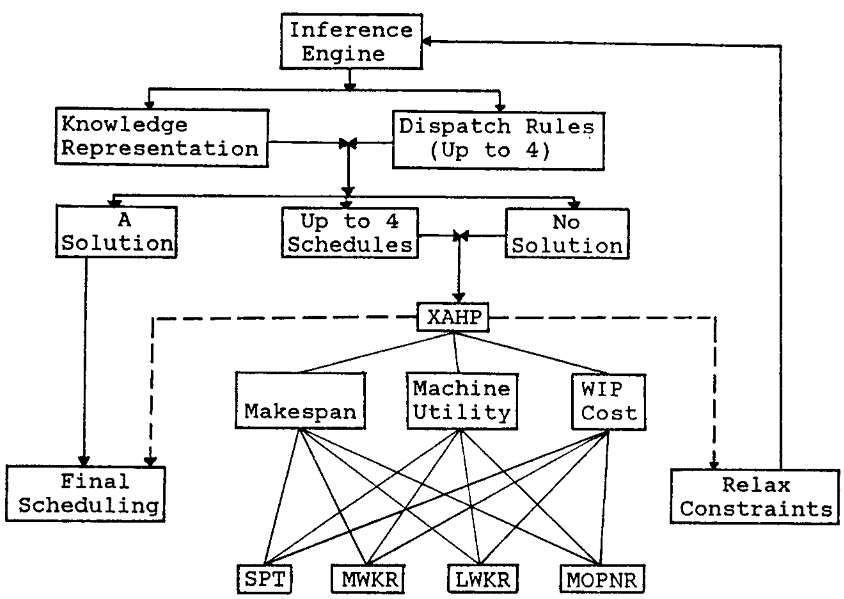


Figure 2: Structure of Knowledge-Based FMS Scheduling with AHP Analysis

(1) Factory environment

```
Unit(scale)
Machine_down(machine, start_time, end_time)
Machine_maintain(machine, start_time, end_time)
```

(2) Current job status

```
Current_job_due date(current job, quantity, due_date)
Current_machine_status(machine, current_job,
     operation_number, end_time, setup_time,
     unit_process_time, cost_in_process)
```

(3) New job status

```
New_job_due date(new_job, quantity, due date)
Machine_route(machine, new_job, operation_number,
     setup_time, unit_process_time, cost_in_process)
Alternative_route(machine, new_job, operation_number,
     setup_time, unit_process_time, cost_in_process)
```

(4) Scheduling

```
Old_schedule(machine, current_job, operation_number,
     start_time, end_time)
New_schedule(machine, job, operation_number, start_time,
     end_time)
```

Each predicate is not only a data structure which stores the information/relation between predicates and components, but also a constraint which can be used by the inference engine to conclude TRUE or FALSE during scheduling. The rules generated by the inference engine are based on the combination of these TRUEs and FALSEs. The following sections discuss detailed search rules (SR). During the backward constraint search process, for every schedulable job, its Jij, MK, and Pijkt are given and stored in the knowledge base. The calculations of MTkt, CTijkt, and STijkt are determined by the following SR's:

SR1: Determining MTkt

```
IF t=1 then
   IF max {Di} in [BSTk, BETk] THEN
      MTkt=BSTk
ELSE
      MTkt=max{Di}
ELSE
      MTkt=STi j k t-1
```

SR2: Determining CTijkt

```
IF t=1 THEN
   CTijkt=MTkt
ELSE
   CTijkt=min{STi j+1 k t,MTkt}
```

SR3: Determining STijkt

```
IF (CTijkt - PTijkt) within [BSTk, BETk] THEN
   STijkt=BSTk
ELSE
   STijkt=CTijkt - PTijkt
```

When the above data are calculated, the next step is to select the best schedulable operation. The selection process involves two possible conditions: conflict-free condition and conflicting condition. The conflicting condition occurs when different jobs' operations in the search space share the same resource. This selection process is designated by the following SR's:

SR4: Best selection condition

```
IF there exists STijkt >= other CTijkt THEN
   select the node whose STijkt is maximum to be scheduled
```

SR5: Rule of preparing conflict

```
IF no best selection condition THEN
BEGIN
   mark the node whose CTijkt is maximum;
   IF more than one choice for maximum CTijkt THEN
   BEGIN
      mark the node whose STijkt is maximum;
      IF more than one choice for maximum STijkt THEN
      BEGIN
              mark the node whose operation has the least
              alternatives;
          IF more than one choice for the least alternatives
              THEN mark the node at random;
      END
   END
END
```

SR6: Rule of conflict-free condition

```
IF no other operation is assigned to the machine of the
   marked node  THEN
   select the marked node to be scheduled.
```

SR7: Rule of Conflicting Condition

```
IF no SR6 exists THEN
   IF the marked node's STijkt >= other CTijkts of the node
      whose the operations are assigned to the machine of the marked node THEN
      select the marked mode to be scheduled
      ELSE
          use the dispatching rule to select a node among
          the nodes sharing the machine of the marked node.
```

SR8: Rule of multiple choices in the use of dispatching rule

```
IF there is more than one choice in the use of the
   dispatching rule THEN
   BEGIN
     select the node whose operation has the least
     alternatives;
     IF more than one choice for the least alternatives
        THEN select the node at random;
   END
```

The above rules depend on the special factory environment and the scheduler's experience. The following SR's describe the generation of the new search space:

SR9: Changing schedulable operation to scheduled operation

```
IF a schedulable operation is selected THEN
BEGIN
   copy its information to predicate new-schedule;
   remove the nodes which have the same selected operation;
   use SR1 to change MTkt to MTk t+1;
END
```

SR10: Generation of new search space

```
IF SR6 is completed THEN
BEGIN
   use SR2 and SR3 to change CTijkt and STijkt to
      CTijkt+1 and STijkt+1;
   IF there exists preceding operation and its alternative
      operations THEN
   BEGIN
      advance t to t+1; insert those operations to new search space at state t+1;
   END
END
```

```
SR11:  Rule of stop search

       IF search space is empty THEN
          stop search
       ELSE
          continue search.
```

Each search rule can be translated to a predicate whose value is either TRUE or FALSE. If the condition is TRUE, the body of statements following THEN will be performed; otherwise, the body of statements following ELSE will be performed. The entire inference engine which conducts backward constraint search is then an iterative process of SR1 through SR11 for each dispatching rule until the best schedule is found.

EXAMPLE OF "SCHEDULE" and "XAHP"

Once a schedule has been established, each job must be assigned to a particular machine to perform the required operations. Two softwares, SCHEDULE and XAHP, are developed to generate the final schedule. The main menu of SCHEDULE (Figure 3) consists of eight choices to perform scheduling functions. The first three choices (Factory Environment Representation, Current Job Representation, and New Job Representation) are the input for scheduling. The input options are from disk or screen. The user enters the filename or picks a filename from a file listing (if inputting from disk). If the user selects the screen input, two windows are displayed and all the necessary information is required.

```
------------------- FMS SCHEDULING -------------------

 1.     Factory Environment Representation ("EMPTY")
 2.     Current Job Representation ("EMPTY")
 3.     New Job Representation ("EMPTY")
 4.     Update Knowledge
 5.     Scheduling
 6.     Save Knowledge
 7.     Remove Knowledge
 8.     End of Scheduling

 Input a number :
```

Figure 3: Main Menu of SCHEDULE System

Choice 3 on the main menu represents current job status, which requires three kinds of information: job name, job quantity, and job due date. For current job routing information, the following information is needed: machine name, job name, operation number, machine free time, setup time, unit process time, and unit cost. To solve the rescheduling problems, a scheduler must input machine free time. If a job can be rescheduled only after a machine has finished all quantities of this job, a scheduler can estimate the information of machine free time according to the time needed to finish the current operation. If an uncompleted job can be rescheduled at any time, the machine free time is zero.

Choice 4 on the main menu updates the knowledge database. Again, two windows are displayed. The top window shows the format for each knowledge representation in computer-readable form. The bottom window is a Word-Star type of word processor which loads the entire knowledge base into the window, allowing a user to modify the knowledge base. When a user completes this function, the system checks the modified knowledge database format to avoid format error.

After the needed information is loaded, a user can select choice 5 from the main menu to perform backward constraint search. Gantt chart, based on the dispatching rules, will be generated from the inference engine if there are feasible solutions. Choice 6 and choice 7 on the main menu save and remove knowledge. Choice 8 exits from the system.

Figure 4 shows the main menu of XAHP. Choice 1 on the main menu executes another expert system which may generate multiple solutions. A user can select choice 1 to execute SCHEDULE system to generate up to four different schedules with four different dispatching rules. SCHEDULE writes the final schedule in the format of "RULE_NAME(value of Makespan, value of Machine utility, value of WIP Cost)" under a file named SCHEDUL.XIN. Choice 2 on the main menu determines which schedule is the best for the current state of FMS environment. Figure 5 displays an example of final selection in AHP analysis.

```
─────────────── XAHP Main Menu ───────────────

    1.  :   Process New Expert System

    2.  :   Postprocess (with AHP)

    3.  :   End of Program

    Please enter a number :
```

Figure 4: XAHP Main Menu

```
RESULTS FOR BOTTOM LEVEL                    MAKESPAN

                             10.00     30.00     50.00     70.00     90.00
                              I....I....I....I....I....I....I....I....I
MWKR(2080,0.32,1365775)       I****                                     I .1667
MOPNR(2004,0.34,1350030)      I*****************************************I .8333
                              I....I....I....I....I....I....I....I....I
                             10.00     30.00     50.00     70.00     90.00

Press <Enter> to continue

RESULTS FOR BOTTOM LEVEL                    UTILITY

                             20.00     35.00     50.00     65.00     80.00
                              I....I....I....I....I....I....I....I....I
MWKR(2080,0.32,1365775)       I****                                     I .2500
MOPNR(2004,0.34,1350030)      I*****************************************I .7500
                              I....I....I....I....I....I....I....I....I
                             20.00     35.00     50.00     65.00     80.00

RESULTS FOR CRITERIA                    RESOLVE ES CONFLICT GOALS

                             30.00     32.50     35.00     37.50     40.00
                              I....I....I....I....I....I....I....I....I
MAKESPAN                      I**************                           I .3333
UTILITY                       I**************                           I .3333
WIP COST                      I**************                           I .3333
                              I....I....I....I....I....I....I....I....I
                             30.00     32.50     35.00     37.50     40.00

Consistency index =   .0000   Consistency ratio =   .0000   Max. eigv =  3.0000


Press <Enter> to continue
```

Figure 5: Result in AHP Analysis

## CONCLUSIONS

In practice, many factories place the responsibility for making job-sequence decisions on the shop floor foreman, where machine tool availability, the specific workman qualifications, and the conditions of machines available at the current scheduling time are considered. In contrast, the top/middle manager levels in a manufacturing organization make decisions on the job due date, job process routing, and company policy. The knowledge of current manufacturing environments at lower levels and organizational decisions at higher levels are sometimes not compromised with respect to the scheduling. The conflicting situation, with regards to schedule/reschedule problems, between upper and lower level departments in a manufacturing organization can be resolved with SCHEDULE and XAHP.

SCHEDULE is the software application orientation used to solve real-time scheduling problems in FMS. It is designed to be implemented at shop floor by the scheduler who is familiar with the required information about the machine assignment. The higher level information related to a scheduling problem can be retrieved from the distributed database in an FMS environment and used in SCHEDULE. XAHP is a system for analyzing multiple solutions from an expert system, and is a tool for determining the final detailed schedule of the current factory environment.

Computer Integrated Manufacturing (CIM) systems are the wave of the future in the United States, especially when facing the competition of low-cost Japanese products. CIM systems are highly automated and consist of computer-controlled machining stations linked by an automated material-handling system. The systems' high productivity is due to their automation and flexibility. The system's scheduling problem is important in that the scheduling must operate in real-time, and must incorporate the availability of resources. The major concern of this paper is how to use the limited resources and resolve the conflict within manufacturing organizations that reflect the FMS environment for a real-time scheduling.

## REFERENCES


1. Adiga, S. & Cochran, J. K. "A Decision Analysis Approach to Modeling Conflicting and Inexact Reasoning in Rule-Based Expert Systems," IEEE System, Man, & Cybernetics, Vol. 1, 983-987, 1985.
2. Bunnag, P. & Smith, S. "A Multifactor Priority Rule for Shop Scheduling Using Computer Search," IIE Transactions, Vol. 17, 141-146, 1985.
3. Dar-El, E. M. & Wysk, R. A. "Job Shop Scheduling: A Systematic Approach," Journal of Manufacturing Systems, Vol. 1, 77-88, 1982.
4. Hershauer, J. C. & Ebert, J. "Search and Simulation Selection of a Job Shop Scheduling Rule," Management Science, Vol. 21, 833-848, 1974.
5. King, J. R. & Spachis, A. S. "Scheduling: Bibliography and Review," International Journal of Physical Distribution and Material Management, Vol. 10. West Yorkshire, England: MCB, 104-117, 1980.
6. Kiran, A. S. & Smith, M. L. "Simulation Studies in Job Shop Scheduling-I A survey," Computers and Industrial Engineering, Vol. 18, 87-93, 1984.
7. Kiran, A. S. & Tansel, B. C. "A Framework for Flexible Manufacturing Systems," Proceedings, IIE Conference, Los Angeles, CA, 37-43, 1985.
8. Morton, M. E. & Smunt, T. "A Planning and Scheduling System for Flexible Manufacturing," A. Kusiak, (Ed.), Flexible Manufacturing Systems: Methods and Studies, Amsterdam, Holland: Elsevier Science, 1986.
9. Saaty, T. L. The Analytic Hierarchy Process, New York: McGraw-Hill, 1983.
10. Shaw, M. J. P. & Whinston, A. B. "Application of Artificial Intelligence to Planning and Scheduling in Flexible Manufacturing," A. Kusiak (Ed.),Flexible Manufacturing Systems: Methods and Studies, Amsterdam, Holland: Elsevier Science, 1986.
11. Steffen, M. S. & Greene, T. J. "A Prototype System for Scheduling Parallel Processors Using Artificial Intelligence Methods," Proceedings, IIE Conference, Vol. 6, 425-433, 1986.
12. Wu, S. Y. D. "An Expert System Approach for the Control and Scheduling of Flexible Manufacturing Cells", Ph.D. Dissertation, Penn. State University, 1987.