INTEGRATING SCHEDULING AND CONTROL FUNCTIONS IN COMPUTER INTEGRATED MANUFACTURING USING ARTIFICIAL INTELLIGENCE

Luis Carlos Rabelo & Sema Alptekin

Computer Integrated Manufacturing Laboratory
Engineering Management Department, University of Missouri-Rolla
Rolla, Missouri 65401

ABSTRACT

Proper integration of scheduling and control in Flexible Manufacturing Systems will make available the required level of decision-making capacity to provide a flexibly-automated, efficient, and quality manufacturing process. To achieve this level of integration, the developments in computer technology and sophisticated techniques of artificial intelligence (AI) should be applied to such FMS functions as scheduling. In this paper, we present an Intelligent Scheduling System for FMS under development that makes use of the integration of two AI technologies -- Neural Networks and Expert Systems -- to provide the intelligence that the scheduling function requires in order to generate good schedules within the restrictions imposed by real-time problems. Because the system has the ability to plan ahead and learn, it has a higher probability of success than conventional approaches. The adaptive behavior that will be achieved contribute to the integration of scheduling and control in FMS.

KEYWORDS: Scheduling, Control, Artificial Intelligence, Neural Networks, Expert Systems, Flexible Manufacturing Systems.

1. INTRODUCTION

The proper integration of scheduling and control activities will provide the necessary capacity to achieve an intelligent and autonomous FMS. Several researchers have proposed models with augmented levels of intelligence for such integration including O'grady and Lee,1 Maley et al.,6 Lin and Chung,1 Gross,4 Chochos and Alami,2 Maimon,5 Wu,12 and Meyer et al.14 All of them have indicated the importance of the role played by the FMS scheduling function. The current literature stresses the utilization of AI to provide an intelligent approach to the FMS scheduling problem to facilitate the integration with control.1,3,4,6,16,17

The FMS scheduling problem has been defined by Kiran and Alptekin47 as follows:

"Given an FMS, its actual state, and a set of parts with known processing requirements and due dates, determine the start and completion time of operations of each part to be produced".

The FMS scheduling problem is a more general case of the job shop scheduling problem42, although several factors make FMS scheduling more complex than classical job shop scheduling:

1. Additional resources constraints;
2. Augmented decision alternatives because of flexibility;
3. Real time operation.

To generate responses on-line, and avoid the combinatorial explosion, an effective problem solving strategy should be utilized that includes real time constraints in order to provide a good solution on time. These qualifications of the FMS scheduling problem involves the utilization of several AI disciplines, such as: data interpretation, feature extraction, distributed decision-making, optimization, and constrained-search. This must justify the use of AI to improve the efficiency and accuracy of scheduling.

In this paper a problem-solving architecture is presented that makes use of hybrid neural networks/expert systems. The architecture of this Intelligent Scheduling System for FMS (ISS/FMS) is shown in Figure 1. ISS/FMS utilizes distributed expert systems (DES), an innovative look-ahead algorithm, and neural networks in order to find a good solution for the FMS scheduling problem in real time.

In the next section of this paper considerations about the integration of neural networks and expert systems will be presented. In the following sections, details about the design of the ISS/FMS are provided. Finally, a summary and possible future enhancements are presented.
2. BACKGROUND

The most common use of AI in scheduling has been the development of expert systems which emulate intelligent behavior. However, these are mainly stand-alone and isolated expert systems. On the other hand, the concept of FMS scheduling (as explained above) is interdisciplinary in nature. Therefore, it is appropriate to integrate several AI technologies in the FMS to form intelligent systems that can meet the requirements. In this paper, the synergy of neural networks and expert systems is used to develop a hybrid system that will increase the level of responsiveness of the scheduling function. This section gives a fundamental overview of how neural networks and expert systems could utilize the best aspects of each in order to create systems more responsive and adaptive to their environment.

2.1 EXPERT SYSTEMS

Expert System technology has been concentrating on the construction of high performance programs in specialized professional domains. This technology involves methods and techniques for development of systems with specialized problem-solving expertise. Expert systems have several prominent characteristics: explanation capabilities, efficient problem-solving strategies, user interface and documentation capabilities, mature technology, and commercially available development tools. However, expert systems have several limitations: inability to learn, domain-dependence, complex knowledge acquisition process, slow execution speed, and inability to handle large data sizes.
2.2 NEURAL NETWORKS

A neural network can be described as an information-processing system composed of a large number of interconnected processing elements (See Figure 2). Each of these processing elements has a number of inputs which are modified by adaptive coefficients (weights) and generates an output signal that is a function of its weighted inputs. The processing elements are often divided into layers—groups of processing elements. Neural networks have several 'natural' characteristics provided by their structure such as: generalization, abstraction, speed, mappings of complex relationships, training by example, and graceful degradation. On the other hand, there are several limitations of neural networks, among them: immature technology, no explanation capabilities, and lack of flexible learning mechanisms.

2.3 ANALYSIS

Expert systems and neural networks might be integrated in order to strengthen the best features of each. This integration will provide systems more responsive and adaptive to the environment. For example, an ES could utilize a neural network to search for the implicit features of a database (hypertext retrieval). In addition, a neural network could be utilized to develop innovative knowledge acquisition strategies, especially when the problem to be solved has a great deal of pattern classification. Also, a neural network might help the expert system problem-solving strategy providing an initial approach as well as support to the user interface.

In the following section, details are given of how the ISS/FMS integrates the nature of the "biologically inspired" behavior of neural networks with the symbolic representation and inference mechanism which characterize expert systems in order to provide good schedules.

3. INTELLIGENT SCHEDULING SYSTEM

FOR FMS/ISS/FMS

The architecture of the ISS/FMS consists of the following subsystems (See Figure 1):

- Expert Scheduler;
- Procedural Programs;
- Neural Network System for Heuristics;
- Expert Look-Ahead Scheduler;
- Neural Network System for Coefficients;
- Learning Unit (Under Development).

The ISS/FMS has been designed using modularity principles in order to facilitate addition of new units to enhance its planning and problem-solving horizon. These units will be explained in the following subsections.

Figure 2. Neural Networks.

3.1 THE EXPERT SCHEDULER

The Expert Scheduler is a rule-based expert system (ES) that utilizes backward chaining. It is able to call several other programs and expert systems. The function of the Expert Scheduler is that of knowledge controller in the ISS/FMS.

In the ISS/FMS, there are declarative, procedural, and "biologically inspired" knowledge. The Expert Scheduler interprets the request for scheduling/rescheduling. After that the Expert Scheduler proceeds to read the files that contain the task database with the following information:

A) Task Code and Job Information. Details of the implementation using predicate logic assertions written in PROLOG are given below:

```prolog
task(Task_Code,Total_Number_of_Jobs),
job(task_code(Task_Code),
job_number(Job_Number),
operations(Number_of_Operations,
(operation(Number,
Machine,Ready_Time,
Process_Time),...))priority(Priority)).
```
B) Constraints imposed by the Shop Level:

1. Performance Measure(s);
2. Constraints imposed by the performance measure;
3. Time in which the result is expected.

The Expert Scheduler reads the files corresponding to the cell status, availability of materials, and according to a time assigned by the upper level, concludes if the generation of a result is possible. If the Expert Scheduler perceives the task to be infeasible, it sends a request to the upper level.

The Expert Scheduler, based on the job data, request data, and degree of feasibility, decides on which heuristic to fire. Also, based on the time frame provided, the Expert Look Ahead Scheduler is fired.

The dispatching rules utilized by the Expert Scheduler are:

SPT: Shortest Process Time; EDD: Earliest Due Date; 
CR: Critical Ratio; SLACK: Slack Time Remaining; S/OPN: Slack/Operation; 
LWR: Least Work Remaining.

The dispatching heuristics are fired according to a set of rules based on statistical analysis of historical data, and the output of a neural network system. The rules based on statistical analysis and those to give specific directions to the neural network system are in two separate modules (See Figure 1). Presently the performance is strictly based on tardiness, but future enhancements will include:

Minimization of In-Process Inventory 
Maximization of Machine Utilization.

3.1.2 Neural Network System for Heuristics

In order to use a neural network, one of the problems to define is the input feature space. One has to develop an appropriate way to represent the problem in order to make it "understandable" for the network. Without this key feature, the neural network will fail to learn the relationship with the efficiency and accuracy desired. For example, for tardiness, we selected the following dimensions for the input feature space (Figure 3).

1. Group Technology.
2. Time Remaining Until Due Date.
3. Number of Jobs.

The job database is formatted to be utilized by the Neural Network System. The output of the network ranks the different scheduling heuristics available. Several neural networks have been trained using the results of simulation runs, taking into consideration the input feature space and ranking of priority rules achieved according to the relative tardiness values. A typical network has 40 input units, 121 hidden units, and 6 outputs. They have been trained using the backpropagation rule.

When the Neural Network System (See Figure 1) receives the prompt from the Expert Scheduler, a set of rules analyzes the job database and decides on which network to utilize. The outputs of the neural network represent the ranking of each heuristic according to the relationship saved in the neural structure. This information is sent to the Expert Scheduler to be used by the controller as explained below.

3.1.3 Controller

The Controller is a rule module in the Expert Scheduler that takes into consideration the confidence factors generated by the rule module based on statistical analysis results and the ranking provided by the Neural Network System. The Controller makes the decision about the specific procedural programs to be called. This decision process determines the best heuristic rule(s) to be used for the problem by evaluating both inputs.

3.1.4 Discriminator

The Discriminator is a rule module in the Expert Scheduler. According to the time frame of the decision making, the discriminator receives the answers to the scheduling problem from the selected heuristics and the Expert Look Ahead Scheduler. It selects the best among them, and proceeds to send the answer to the global database of the system. If the final schedule does not meet some of the high priority constraints, the Controller checks the decision time frame to determine the time constraints. It then makes changes to the job database and recursively proceeds with the process.
3.2 LOOK AHEAD SCHEDULER

This expert system implements a new feedback based heuristic scheduling procedure. This heuristic procedure which was utilized mainly for the tardiness criterion, has been proven to be efficient for the FMS scheduling problem (See Table 1). A neural network system is utilized to provide certain coefficients to accelerate the performance of the algorithm and improve its efficiency. Also, it implements heuristics in order to optimize the solution achieved if the decision time frame permits.

The Feedback Heuristic of Kiran and Alptekin is based in a job priority according to:

$$\text{Priority} = (1-\alpha) \times \text{Priority}_{old} + \alpha \times \text{Tardiness}$$

where $\alpha$ and $\beta$ are two coefficients with $0 < \alpha < 1$ and $0 < \beta < 3$.

Table 1: Average Tardiness Performance of the heuristics for FMS problems

<table>
<thead>
<tr>
<th>Number of Machines</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha coefficient</td>
<td>6.6</td>
<td>7.8</td>
<td>9.0</td>
<td>10.2</td>
</tr>
<tr>
<td>Beta coefficient</td>
<td>0.4</td>
<td>0.6</td>
<td>0.8</td>
<td>1.0</td>
</tr>
</tbody>
</table>

3.2.1 Considerations to the Feedback Heuristic

There are important considerations to make about the Feedback Heuristic scheduling procedure. These considerations are necessary because they provide the justifications for using AI techniques in order to improve the performance of this algorithm.

Coefficients

The relationship between the job characteristics and the coefficients is very complex. The coefficients determine an optimal solution for the scheduling problems. It is possible to generate several alpha and beta pairs and after that select the best among them. But this might be computationally expensive or against the real time control considerations (see Table 2). Therefore, it is necessary to provide a way to determine the coefficient from the job database.

Heuristics to Search for Coefficients

We have developed our own heuristics to work from results of a set of pairs of alpha and beta to get a better solution. These heuristics control the strategy of getting new pairs of coefficients, and also when to stop searching because of the low probability of generating better results.

3.2.2 Neural Network System For Coefficients

A neural network, with all its advantages of learning without programming, and software available for fast prototyping with information about 120 pairs of coefficients, will perform the mapping of the relationship and provide a solution that could be evaluated faster than any set of rules. Also, the probability of generating an efficient pair of coefficients will be increased.

Several prototypes were designed around the backpropagation neural network paradigm. This paradigm has been proven to be able to make complex mapping of relationships.

A special editor was developed that took 150 examples and formatted the input and the respective output. It was then decided to generate the different prototype structures. Finally, each one of the structures was evaluated on the following parameters:

1. Iterations needed;
2. Learning performance;
3. Learning time.

The neural network selected had 10 inputs, 51 hidden units (one hidden layer), and 22 outputs. It was able to map the relationship after 20,000 iterations. Scheduling problems out of the training set were presented to the network, and it was able to predict the optimum alpha and beta value in 90 percent of the cases. After evaluation of the results the following modifications were implemented:

Table 2: Sensitivity of Coefficients for a specific job database.
Increase the input units to 40;
Increase the hidden units to 225;
Increase the number of training examples to 300.

With this structure the computational cost was reduced and the performance of the algorithm provided significantly better results.

4. SUMMARY AND FUTURE ENHANCEMENTS

The design of ISS/FMS has been the product of several technologies. The key was the hybrid neural network / expert system approach.

The addition of the learning unit to the system is projected in order to provide a way to check whether the different rule-based and neural networks have been effective. If not, changes will be advised and re-teaching sessions will be automated in order to improve the system and make it more responsive to the environment. A tandem expert system for the learning unit with the backpropagation learning rule, a set of rules to add new hidden units to a hidden layer, and modified training sets to improve the performance and responsiveness of the system are under development. This unit will provide on-line learning. This additional function will increase the level of responsiveness of the scheduling function. And, hence the strength necessary to integrate scheduling and control functions in FMS.

REFERENCES