Solve scheduling problems with a fuzzy approach

Wang, H.G.; Rooda, J.E.; Haan, J.F.

Published in:

Published: 01/01/1996

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 20. Jan. 2019
Solve Scheduling Problems with a Fuzzy Approach

H.G. Wang, J.E. Rooda and J.-F. Haan
Systems Engineering Division
Department of Mechanical Engineering
Eindhoven University of Technology
P.O.Box 513, 5600 MB Eindhoven
The Netherlands
{h.g.wang, rooda}@wtb.tue.nl

Abstract

Scheduling is a complex problem which occurs often in a manufacturing environment. Most of the available schedulers are based on a simulation approach using dispatching rules. These rules are often dedicated to the satisfaction of a single performance criterion, and are used whatever the characteristics of the system or the type of jobs [1], though scheduling is a multi-criteria problem. In this paper the scheduling problems are addressed using a non-classical approach supported by fuzzy control theory. The ability to deal with multivariables makes fuzzy control a good alternative for the scheduling problems because it can easily make compromises among multi-criteria by properly combining elementary dispatching rules. These compromises can easily be adjusted in accordance with the objectives of the system and the characteristics of the jobs. The proposed method has been implemented and tested in a discrete event simulation environment. The results are presented in this paper.

1. Introduction

Scheduling is to properly allocate resources over time to perform a collection of tasks. The practical problem of scheduling arises in a variety of situations. For manufacturing systems that are responsible for manufacturing and transporting products, there are unavoidable waiting queues in front of machines because of the limited capacity of the machines and other factors, like technology and social conditions. The lead time of an order is often several times (5 to 10 times) of the processing time [2]. The control of the waiting queue or the waiting time by means of scheduling to meet various criteria is an important aspect of production control.

There are generally two kinds of major requirements of a manufacturing system: the customers' satisfaction and the resources utilization's optimization. In practice there are plenty dispatching or priority rules being used for ordering the jobs in a queue in front of a machine. Each rule intends to satisfy a special criterion. The often encountered criteria are, for instance, lateness or tardiness minimization and lead time minimization.

Since a manufacturing system is always a multiobjectives or multi-criteria system, it is important to have a way to easily make compromises among elementary criteria. A lot of work has been done to achieve such a compromise by means of the so-called aggregated rules which can provide intermediate result of the elementary rules of which they are composed [3, 4, 5]. But there is a problem of easily adapting these aggregated rules which are necessary when the manufacturing system changes. For a recent survey of dispatching and aggregated rules we refer to [6].

Work has been carried out to make the scheduler more flexible and adaptable by means of advanced technologies, like fuzzy control and neural networks [1, 4, 7, 8]. In this paper we present a new approach to make a compromise by using fuzzy control theory. The balance of each elementary dispatching rule can be realized by modifying the fuzzy decision rule base. After a brief introduction of the suggested approach in the next section, the model which will be employed for our simulation study is illustrated in Section 3. The simulation results will be presented and analyzed in Section 4. Finally in Section 5 some conclusions are drawn and suggestions for further research are presented.
2. Fuzzy scheduling approach

The fuzzy scheduling approach provides a way to combine different dispatching rules with potential adapting ability. It doesn't depend on what rules will be chosen. According to Grabot et al. there are three common applied rules in present scheduling, namely shortest processing time (SPT), slack time or earliest due date (EDD) and priority rules. We shall emphasize in this paper on the compromise between SPT and EDD rules on the basis of reasons in [1, 9].

Before we go on with the approach itself, we shall give a brief introduction on fuzzy control theory. The fuzzy control concept is based on the fuzzy set theory, which was introduced by Zadeh in 1965. This theory was introduced to represent and to reason with fuzzy or linguistic concepts and is able to tackle several interrelated linguistic variables. For more information on fuzzy set theory, we refer to [11, 12].

![Fuzzy Controller](image)

**Figure 1: The fuzzy control structure**

The structure of a fuzzy controller is usually that of Figure 1. It consists of four elements, namely the Fuzzification (N/F), the Inference Engine (IE), the Defuzzification (F/N) and the Knowledge Base (KB). The function of each element has already been described in [3], while a brief introduction of each element is presented here for readers' convenience. S is the controlled system.

In this study the input variables sent to the fuzzy controller are processing time (p) and due date (d) respectively. The first step is to represent these two variables symbolically, and to associate fuzzy sets or membership functions with them as shown in Figure 2. XS, S, N, L, XL means very short, short, normal, long, very long; XC,C,N,F,XF means very close, close, normal, far, very far; and XL, L, M, H, XH means very low, low, medium, high, very high. The N/F will transform the numerical input values into fuzzy values via a mapping process. With observation $p_0$ for processing time (Figure 2(a)), the corresponding fuzzy values are $\mu_{xs}$ and $\mu_s$ in the XS set and the S set, respectively. With the observation $d_0$ for due date (Figure 2(b)), its fuzzy value is $\mu_{xc}$ in the XC set. These fuzzy values represent to what degree the observed value belongs to each set associated with the variable. The reasoning process then evaluates the decision rules with the fuzzy values obtained. Mamdani's implication and inference are employed for the reasoning process in this study. The hatched area of priority (Figure 2(c)) is the combined fuzzy set obtained by evaluating two relevant rules in the fuzzy decision rule base (Table 1). The first row is the five sets associated with processing time $p$ and the first column is the five sets associated with due date $d$. The two relevant rules are: if $p$ is XS and $d$ is XC then $pr$ is XH and if $p$ is S and $d$ is XC then $pr$ is H. The reasoning process is carried out by the IE.

<table>
<thead>
<tr>
<th>Rule Base 1</th>
<th>XS</th>
<th>S</th>
<th>N</th>
<th>L</th>
<th>XL</th>
</tr>
</thead>
<tbody>
<tr>
<td>XS</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>XC</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>C</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>N</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>F</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>XF</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule Base 2</th>
<th>XS</th>
<th>S</th>
<th>N</th>
<th>L</th>
<th>XL</th>
</tr>
</thead>
<tbody>
<tr>
<td>XS</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>XC</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>C</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>N</td>
<td>H</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>F</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>XF</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
</tbody>
</table>

The fuzzy values obtained by the reasoning process will be used as ordering priority after the defuzzification process. The defuzzification process, performed by F/N, translates the fuzzy values into crisp or numerical values, $pr_0$. The membership functions which are needed for the fuzzification and defuzzification processes are stored in the KB. The decision rules which are derived from the knowledge about elementary dispatching rules are also stored in the KB.

The method suggested in [1] to balance different elementary dispatching rules is to assign a weighting factor for each elementary rule and to specify the weighting factors according to the importance of each rule in the aggregated rule. The balance can be more intuitively and easily realized in the rule base itself. Take
the diagonal from the right-top to the left-bottom as reference. If we shift the items up or down along this diagonal the relative importance of each elementary dispatching rule in the aggregated rule will be changed. In Table 2, the due date rule become more important comparing with that in Table 1 by shifting some of the items up along the diagonal. Thus the processing time rule become relatively less important in the aggregated rule. In principle we can modify each item in the rule base to balance the two elementary dispatching rules with different importances, if the rule base satisfies the constraints given in [ll]. These two rule base will be employed in the simulation study later in this paper.

3. Simulation model

For simulation there must be a model of the system. A descriptive model of a simple system is realized using a formalism for the description of parallel cooperating process. The model consists of a number of processes and channels connecting these processes [14]. A job-shop (JS) together with its environment, namely customers (Cu) and suppliers (Su), is shown in Figure 3. Customers ask for various products from the job-shop by means of orders. Each product follows a given process plan or a recipe that specifies the sequence of machines it must visit and the operations performed by these machines. Suppliers are responsible for supplying the required material to the job-shop.

![Figure 3: Job-shop with its environment](image)

The manufacturing process is taken place in a job-shop. The job-shop consists of controller (JSC), store (JSS), transporter (JST) and three different workstations (WS1, WS2, and WS3) as illustrated in Figure 4. Controller JSC is not only responsible for the good communication with customers and suppliers, but also takes care of the monitoring and controlling the real production process.

![Figure 4: The simple job-shop model](image)

A workstation is presented in Figure 5. There are unavoidable waiting queue of jobs in front of machine (M). The waiting queue is often modelled in the form of a buffer (B). The buffer has a limited capacity, so the waiting queue has a limited length. This queue is ordered on the basis of priorities that are specified by the fuzzy scheduler (FS). Buffer B sends processing time and due date information to the fuzzy scheduler, and the fuzzy scheduler will make a decision on the priority of each job following the aforementioned approach. The machine keeps on picking up the first job in the queue to process.

4. Simulation study

A series of simulation experiments have been performed with the model to investigate the scheduling performance. The abbreviations mlt, ml, mt and nlo
are used to represent the mean lead time, mean lateness, mean tardiness and number of late orders respectively. The unit for mlt, ml and mt are hours. The first series concerns how many fuzzy sets should be associated with each of the input and output variables. The more the sets are chosen, the bigger the fuzzy decision rule base will be. It is expected that better performance can be obtained with a big rule base, but with an enormous complexity. The results are presented in Figure 6. When three sets are associated with each variable, the shortest mlt is obtained with the biggest ml. The results with 5 sets and 7 sets are better in the sense that they provide a good compromise between the two criteria. Since there is only a trivial difference in scheduling performance between the 5 sets and 7 sets case, we shall use the 5 sets case for the simplicity reason in our further study.

Figure 5: A workstation model

Figure 6: Experiment on chosen the number of fuzzy sets

Figure 7 presents the results of a comparison between the fuzzy aggregated rule and the corresponding elementary dispatching rules. Shortest processing time (SPT) and earliest due date (EDD) rules are the two elementary dispatching rules employed in this study. Fuzzy shortest processing time and fuzzy earliest due date rule (FPFD) is the fuzzy aggregated rule. It can easily be observed that the fuzzy aggregated rule provides intermediate performance comparing to that of the elementary rules on which it composed. The results confirm that the fuzzy approach is indeed a good alternative for scheduling to achieve multi-objectives.

Figure 7: Comparison of fuzzy aggregated rules with elementary rules

To show the ability to adapt the scheduling compromise by means of adjusting the fuzzy decision rule base, simulation experiments have been run with two different rule bases as aforementioned in Section 2. From the simulation results, refer to Figure 8, we can see that they confirm what we have expected. The due date performance by applying rule base 2 is better than that by applying rule base 1. This kind of easy adaptable scheduler is a promising alternative for solving scheduling problems. The method doesn't depend on what elementary scheduling rules will be used. So any two elementary rules can be aggregated and adapted in a way as suggested in this paper.
5. Conclusions and recommendations

The optimization of the resources’ utilization is an often encountered problem in systems, which requires the use of scheduling technique. The suggested scheduling approach presented in this paper illustrates how fuzzy control concepts in a pragmatic and direct manner make the compromise between different dispatching rules possible. The combined rules provide intermediate performance of the system comparing the elementary rules on which it composed. The balance of the elementary rules via the fuzzy decision rule base adjustment shows the flexibility of the scheduling approach. Any individual elementary dispatching rules can be combined by using this approach. Further study should be carried out on the comparison of the classical and fuzzy aggregated rules, and on exploring a systematic procedure to design a fuzzy scheduler. A three or more criteria compromise by combining three or more elementary dispatching rules also deserves further study.

References


