JOB SHOP SCHEDULING WITH LOT-SIZING AND BATCHING IN AN UNCERTAIN REAL-WORLD ENVIRONMENT

Sanja Petrovic1, Carole Fayad1 and Dobrila Petrovic2

1School of Computer Science & IT, University of Nottingham, Jubilee Campus, Wollaton Road, Nottingham NG8 1BB, UK, {sxp,cxf}@cs.nott.ac.uk
2School of Mathematical and Information Sciences, Coventry University, Priory Street, Coventry CV1 5FB, UK, {D.Petrovic@coventry.ac.uk}

Abstract: This paper proposes a novel algorithm for a real-world job shop-scheduling problem, where both lot-sizing and batching processes are considered. A fuzzy rule-based system is developed for determining lot sizes, where the input variables are workload on the shop floor, size of the job and its urgency while the output is the size of the lots. Both input and output variables are modelled as linguistic variables with imprecise values represented by using fuzzy sets. A fuzzy multi-objective genetic algorithm is developed to generate schedules of jobs whose processing times and due dates are imprecise and modelled by using fuzzy sets. A genetic algorithm takes into consideration the determined size of lots for jobs, and considers batching together jobs of similar characteristics in order to reduce the required setup time. The objectives considered are to minimize average tardiness, number of tardy jobs, setup times, idle times of machines and throughput times of jobs. The developed algorithm is tested on real-world data obtained from a printing company.

Key words: Job shop scheduling, lot-sizing, batching, fuzzy inference system, fuzzy multi-objective genetic algorithm, real-world application

1. INTRODUCTION

A job shop scheduling problem is an optimisation problem in which \( N \) jobs have to be allocated to \( M \) machines. Each job is assigned a predefined order of processing on the machines. Job shop scheduling problems attracted a considerable interest of the scheduling community and consequently the literature on this topic is rather rich (Pinedo, 2002; Yamada and Nakano, 1997, to mention just a few). In spite of this, there is still a gap between the academic research on job shop scheduling and real-world problems.

In this paper, we consider two issues within job shop scheduling and these are batching and lot-sizing that are important in many real-world scheduling problems. Batching refers to the decision of scheduling similar jobs contiguously, while lot-sizing deals with the decision on when and how to split a job into lots (Potts, 1995; Yavuz, 1999).

Batching is of particular importance in manufacturing systems in which changing of type of product to be processed on a machine incurs a setup time/cost. A set of jobs is grouped into families on the basis of their production requirements. As a result, no setup is needed when jobs of the same family are scheduled.
consecutively. However, scheduling of jobs that belong to different families, one after the other, incurs setup time/costs.

The size of a batch appears to be a sensitive issue. Large batches improve the efficiency of the production line as they save time and also increase the machine utilization. On the other hand, a batch may group jobs of different priorities, and consequently, jobs of high priorities can be processed in different batches. Therefore, a customer service may be improved by having smaller batches. The recent survey of Potts and Kovalyov (2000) gives the extensive literature on models that integrate scheduling with batching decisions. They review models of batch scheduling developed for all classical machine settings. Agnetis et al. (2004) deal with the problem of finding a schedule that minimises the number of setups when jobs are subject to precedence constraints.

The concept of lot-sizing is introduced to partially satisfy customer’s demand and to produce and deliver, in the first instance, enough amount of a product so that the customer is not out of stock of requested product while awaiting the whole job to be completed. The remaining part of the job will be delivered at some later date. In job-shop scheduling applications, the creation of lots permits the overlapping of different operations of the same job on parallel machines and may therefore also reduce throughput time (Potts, 1995). The sizes of lots have to be carefully chosen in order to avoid a schedule with a large number of lots of very small size that need large setup time on machines. The initial research on lot-sizing was related to the Economic Order Quantity (EOQ), where demand for a product is assumed to be stationary, i.e., an order for a product is continuously placed with a constant rate of demand within an infinite planning horizon. Economic Lot Scheduling problem (ELSP) takes into consideration capacity constraints (for example, machines capacity, inventory space capacity, etc.). A quite major step was made by introducing dynamic demand conditions. The Wagner-Whitin problem was introduced, which subdivided the finite planning horizon into discrete periods (Drexl and Kimms, 1997). Current approaches to lot-sizing alternate between capacitated (Sarker and Newton, 2002), discrete (Ouenniche and Boctor, 1998) or a combination of these two approaches (Drexl and Kimms, 1997). Karacapilidis et al. (2000) introduced a fuzzy methodology for lot-sizing of multiple products in a production system, where demand for a product is expressed by linguistic terms.

The research described in this paper investigates batching and lot-sizing problems in a printing company Sherwood Press, in Nottingham, UK. Decisions regarding batching and lot-sizing are made in the presence of uncertainty. Processing times of jobs are uncertain due to both machines and human factors, while due dates are also flexible and allow the decision maker (scheduler) to express his/her attitude toward the tardiness of jobs. Processing times and due dates of the jobs are modelled by fuzzy sets. Through the series of interviews with Sherwood Press staff, variables identified to be important for lot-sizing are workload on the shop floor, job size and job urgency. In order to mimic the way that the scheduler decides on the lots and lot-sizing, fuzzy IF-THEN rules are introduced which derive conclusions based on imprecise premises. A genetic algorithm (GA) is developed which takes into consideration uncertain processing times and due dates of jobs and the decision on lot-sizing of jobs derived using the fuzzy rules. A number of objectives are used simultaneously during the GA search to measure the quality of the schedules. These objectives are average tardiness, number of tardy jobs, total setup time, total idle time of machines and total throughput time of jobs.

The paper is organized as follows. Section 2 introduces a job shop problem and the real-world instance of the problem present at Sherwood Press. Section 3 describes fuzzy IF-THEN rules developed for lot-sizing, while a fuzzy GA for job shop scheduling is introduced in Section 4. Section 5 discusses the results obtained on real-world data obtained from Sherwood Press, followed by conclusions.

2. PROBLEM STATEMENT

The notation used in the problem statement and throughout the paper is as follows.

\[ N \] total number of jobs
Job Shop Scheduling with Lot-Sizing and Batching in an Uncertain Real-World Environment

\[ J_j \quad \text{job, } j=1, \ldots, N \]
\[ M \quad \text{total number of machines} \]
\[ M_i \quad \text{machine, } i=1, \ldots, M \]
\[ F \quad \text{total number of families of jobs} \]
\[ F_j \quad \text{family of jobs, } j=1, \ldots, F \]
\[ H \quad \text{total number of unit time periods of the planning horizon} \]
\[ t \quad \text{period in the planning horizon, } t=1, \ldots, H \]
\[ q_j \quad \text{quantity of identical items in job } J_j, j=1, \ldots, N \]
\[ r_j \quad \text{release date of job } J_j, j=1, \ldots, N, \text{ which denotes when a job can start its processing} \]
\[ s_j \quad \text{start date of job } J_j, j=1, \ldots, N, \text{ which denotes when a job has started its processing} \]
\[ d_j \quad \text{due date of job } J_j, j=1, \ldots, N \]
\[ C_j \quad \text{completion date of job } J_j, j=1, \ldots, N \]
\[ C_{max} \quad \text{makespan } C_{max}=\max(C_1, \ldots, C_N) \]
\[ T_j \quad \text{tardiness of job } J_j, j=1, \ldots, N \]
\[ T_j = \max \big( C_j - d_j, 0 \big) \]
\[ w_j \quad \text{importance of job } J_j, j=1, \ldots, N \]
\[ (i,j) \quad \text{operation of job } J_j, j=1, \ldots, N \text{ processed on machine } M_i, i=1, \ldots, M \]
\[ p_{ij} \quad \text{processing time of operation } (i,j) \text{ given in unit time periods, } i=1, \ldots, M, j=1, \ldots, N \]
\[ L_j \quad \text{number of lots of job } J_j \]
\[ (i,j,l) \quad \text{an operation } (i,j) \text{ of lot } l, i=1, \ldots, M, j=1, \ldots, N \text{ and } l=1, \ldots, L_j \]
\[ q_{ijl} \quad \text{number of identical items of an operation } (i,j,l), \]
\[ i=1, \ldots, M, j=1, \ldots, N \text{ and } l=1, \ldots, L_j \]
\[ p_{ijl} \quad \text{processing time of an operation } (i,j,l), i=1, \ldots, M, j=1, \ldots, N \text{ and } l=1, \ldots, L_j \]
\[ p_{ijl} = \frac{p_{ij} \cdot q_{ijl}}{q_j} \]
\[ C_{ij} \quad \text{completion time of an operation } (i,j), i=1, \ldots, M, j=1, \ldots, N \]

The job shop scheduling problem considered in this research consists of \( N \) jobs \{ J_1, \ldots, J_N \} with given release and due dates \( r_j, d_j, j=1, \ldots, N \), respectively, that have to be scheduled on a set of \( M \) machines \{ M_1, \ldots, M_M \}. Each job consists of a chain of operations on which precedence constraints are imposed. Each operation of job \( J_j \) processed on machine \( M_i \) is represented by an ordered pair \((i,j)\), while
its processing time is denoted by $p_{ij}$. Each job belongs to a certain family. Families of jobs are determined on the basis of the colour requirements of jobs.

The scheduling problem is formulated as follows. Find a non-preemptive sequence of operations of $N$ jobs on each of $M$ machines taking into consideration the following objectives:

(I) to minimise the average tardiness of schedule $s$

$$AT(s) = \frac{1}{N} \sum_{j=1}^{N} w_j T_{wj}$$ (1)

(II) to minimise the number of tardy jobs in schedule $s$

$$NT(s) = \sum_{j=1}^{N} u_j, \text{ where } u_j = \begin{cases} 1 & \text{if } T_j > 0 \\ 0 & \text{otherwise} \end{cases}$$ (2)

(III) to minimise the total setup time in schedule $s$

$$ST(s) = \sum_{i=1}^{M} \sum_{f=1}^{F} a X_{fi},$$ (3)

where $X_{fi} = \begin{cases} 1 & \text{if there is a need for a setup for a family of jobs at period } t \\ 0 & \text{otherwise} \end{cases}$

$X_{fi}$ is referred to as the changeover coefficient and parameter $a$ has the value of 20 minutes, which is the time needed to make a machine ready for a changeover of the job families.

(IV) to minimise the total idle time of machines in schedule $s$, where the idle time is defined as time within the planning horizon during which the machines are used neither for processing of jobs nor for setup

$$IT(s) = HM - \left( \sum_{i=1}^{M} \sum_{f=1}^{F} a X_{fo} - \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{f=1}^{F} p_{ij} \right)$$ (4)

(V) to minimise the total throughput time defined as the total time that jobs spend on the shop floor in a schedule $s$

$$TT(s) = \sum_{j=1}^{N} (C_j - s_j); \quad j = 1, ..., N$$ (5)

2.1 Real-world Job Shop Problem

A job shop problem present in Sherwood Press involves a number of machines ($M=18$), which are grouped into 7 work centres: Printing, Cutting, Folding, Card-inserting, Embossing and Debossing, Gathering, Finishing (Stitching, Trimming and Packaging). Each job (a printing material) has a predetermined order of processing on machines in some or all work centres. Processing times of jobs are uncertain and represented by fuzzy sets. The definition of fuzzy sets is given in the Appendix.

The estimation of processing time of each operation is obtained taking into consideration the nature of the machine in use. While some machines are automated and can be operated at different speeds, others are staff-operated and therefore the processing times are staff-dependent. Uncertain processing times $\tilde{p}_{ij}$ are
modelled by triangular membership functions represented by a triplet \((p_{ij}^1, p_{ij}^2, p_{ij}^3)\), where \(p_{ij}^1\) and \(p_{ij}^3\) are lower and upper bounds of the processing time, while \(p_{ij}^2\) is a so-called a modal point (Klir, 1998). An example of fuzzy processing time is shown in Figure 1.

Following the scheduling practice in Sherwood Press, we classify jobs into three groups on the basis of their priorities and the corresponding tolerance of their tardiness: High, Medium and Low. A job of High priority is expected to be processed on time with no delay, and therefore there is a crisp due date defined for it. A job of Medium priority has a due date that can be extended with two additional days, whereas a job of Low priority is given a tolerance of delay of a maximum of one week on top of the originally set due date. The imprecise due dates \(\tilde{d}_j, j = 1, \ldots, N\), are represented by trapezoidal fuzzy sets illustrated in Figure 2.

3. FUZZY LOT SIZING

In order to deliver at least part of the job ordered on time, jobs are split into lots. Three variables are identified to be important for the decision on lot-sizing: the workload on the shop floor, size of the job and urgency of the job with respect to its due date. Fuzzy IF-THEN rules are defined to decide on the lot-sizing using imprecise values of these three variables. Workload takes into consideration the number of items to be processed on the shop floor and is described by two linguistic terms: Not Large and Large. Urgency of the job is calculated as the difference between its due date and release date and is described by three linguistic terms: High, Medium and Low. Very Small, Small, Medium and Large are the linguistic terms used to describe size of the job. These membership functions are defined in collaboration with Sherwood Press following their policy for lot-sizing. They are given in Figure 3.

Fuzzy IF-THEN rules are defined based on the policy of Sherwood Press regarding lot-sizing. In order to keep the number of required setups as low as possible, jobs are split into maximum two lots. Therefore, the purpose of the fuzzy IF-THEN rules is to decide whether to split a job into lots and on the size of each lot which is given as percentage of the size of the job.

It would be very difficult to define all 24 fuzzy rules, which determine the size of lots for all possible values of the three variables (workload, size of job, urgency of job) in the premises of the rules. To overcome this difficulty, we define fuzzy rules for each of the three variables independently and apply them in sequence (Petrovic, 1991). A conclusion variable change of lot is introduced in the fuzzy rules, which indicates a necessary change of the current size of the lots. It is described by linguistic values Large Negative, Medium Negative, Small, Medium Positive, Large Positive. The membership functions of the corresponding fuzzy sets are given in Figure 4.

The starting point in the lot-sizing decision is to split each job into two lots of equal size, i.e., 50% of the job size. A set of fuzzy rules is defined which considers workload, size of jobs and urgency of jobs independently in order to derive a change of lot. The final size of the two lots is obtained as an aggregated result of firing of all the rules. Fuzzy rules are given in Figure 5.

The rules attempt to reflect a policy of the company. For example, if the workload is Large, then the idea is to have the first lot of smaller size and to postpone the processing of the major part of the job. Similar holds true for Large jobs. Opposite to that, smaller jobs are processed as much as possible in the first lots. On the other hand, if a job is of High urgency it should be processed completely as soon as possible, i.e., the first lot should be rather large, while jobs of Low urgency will be left mostly for processing in the second lots.

A fuzzy rule based system developed for lot-sizing comprises four components (Negnevitsky, 2002):

1. A fuzzifier takes crisp input values and determines the degrees to which they belong to fuzzy sets in the premises of the rules.
2. A fuzzy rule base given in Figure 5 contains rules with one imprecise variable in the premises of the rules (workload, size of job or job urgency) and imprecise conclusion (change of lot).

Generally, fuzzy rule $R_s$ with one premise and one conclusion has the following form:

$$R_s: \quad \text{IF } x \text{ is } A_s \text{ THEN } y \text{ is } B_s$$

where $x$ and $y$ are imprecise variables, $A_s$ and $B_s$, $s = 1, \ldots, S$ are fuzzy sets, $S$ is the total number of rules.

3. A fuzzy inference engine derives imprecise conclusions based on imprecise premises. The truth-value of a rule premise is equal to the value of the membership function of its premise $\mu_{A_s}(x_0)$ for a given crisp input $x_0$.

The truth-value of the conclusion is determined using the cutting (or truncation) method, where the membership function of the fuzzy set in the conclusion of the rule is cut at the level of the obtained truth-value $\mu_{A_s}(x_0)$ of the corresponding premise. The obtained fuzzy set is denoted by $\mu_{A_s \rightarrow B_s(x_0)}(y)$ and has the following membership function:

$$\mu_{A_s \rightarrow B_s(x_0)}(y) = \min\{\mu_{A_s}(x_0), \mu_{B_s}(y)\}$$

A final conclusion $R$ is a fuzzy set obtained by aggregating conclusions of all the rules that have fired. Maximum operator is used for aggregation as follows:

$$\mu_R(y) = \max \{\mu_{A_1 \rightarrow B_1(x_0)}(y), \ldots, \mu_{A_S \rightarrow B_S(x_0)}(y)\}$$

4. A defuzzifier maps a final fuzzy set obtained by the fuzzy inference engine into a crisp value $\bar{y}$, which represents the fuzzy set “the most suitable” and “correctly”. Our system uses a centre of gravity which is often applied defuzzifier:

$$\bar{y} = \frac{\sum_{s=1}^{S} y_s \cdot \mu_R(y)}{\sum_{s=1}^{S} \mu_R(y)}$$

4. A FUZZY GENETIC ALGORITHM FOR JOB SHOP SCHEDULING

A genetic algorithm (GA) is an iterative search procedure widely used for solving combinatorial optimisation problems, motivated by biological systems and natural genetics (Reeves, 1995). The main characteristics of the GA developed for job shop scheduling with batching and lot-sizing are described below (further details on the GA are given in Fayad and Petrovic (2005)).

Each chromosome consists of two sub-chromosomes of length $M$ (number of machines), named machines sub-chromosome and dispatching rules sub-chromosome. The genes of the first sub-chromosome contain machines, while genes of the second sub-chromosome contain the dispatching rules to be used for sequencing operations on the corresponding machines. The machine sub-chromosome is filled in by randomly choosing machines $M_i$, $i=1, \ldots, M$. Dispatching rules sub-chromosome is initialised by choosing randomly one among the following six rules: Early Due Date First, Shortest Processing Time First, Longest Processing Time First, Longest Remaining Processing Time First, Highest Priority First, Same Family of Jobs Together. While the first four dispatching rules are well-established and widely used in the literature on job shop scheduling (Pinedo, 2002), the last two are tailored to this application with the aim to reduce throughput time of jobs of higher priority and to schedule, whenever possible, jobs of the same families contiguously.

The fitness function is used to evaluate the quality of a given schedule driving the search towards promising areas. In our problem, objectives are measured in different units i.e., they are incommensurable. For example, average tardiness of jobs are given in time units, while the number of tardy jobs takes an integer value from the interval $[0, N]$. However, the objective values have to be used simultaneously to assess the quality of schedules. Satisfaction grades are introduced for each objective to reflect the decision maker preferences with respect to the achieved values of the objectives. Values of the objectives are mapped
into satisfaction grades, which take values from the interval \([0, 1]\), where 0 represents full dissatisfaction and 1 full satisfaction with the achieved objective value. The satisfaction grades with the achieved values of all the objectives are combined in an overall satisfaction grade.

Fuzzy processing times of job operations imply fuzzy completion times of jobs. Before we can assess the satisfaction grades of objectives that are completion and due date related (given in (I) and (II)), we need to establish a way to compare a fuzzy completion time of a job with its fuzzy due date and therefore calculate the likelihood that a job is tardy. We use the method of area of intersection introduced by Sakawa (2000) which measures the intersection between the two fuzzy values \(\tilde{C}_j\) and \(\tilde{d}_j\) (Figure 6). The satisfaction grade of a fuzzy completion time of job \(J_j\) is defined in the following way:

\[
S_{\tilde{C}j}(\tilde{C}_j) = \frac{(\text{area } \tilde{C}_j \cap \tilde{d}_j)}{(\text{area } \tilde{C}_j)}
\tag{9}
\]

The objectives (I) to (V) are transformed into the objectives to maximize the corresponding satisfaction grades as follows.

(I) Satisfaction grade of Average Tardiness \(S_{AT}\)

\[
S_{AT} = \frac{1}{N} \sum_{j=1}^{N} S_{T}(\tilde{C}_j)
\tag{10}
\]

(II) Satisfaction grade of Number of Tardy Jobs \(S_{NT}\) is presented in Figure 7

\[
S_{NT} = \begin{cases} 
\frac{1}{(Max_{NT} - nTardy)/Max_{NT}} & \text{if } nTardy = 0 \\
0 & \text{if } 0 < nTardy < Max_{NT} \\
0 & \text{if } nTardy \geq Max_{NT}
\end{cases}
\tag{11}
\]

where \(nTardy\) is the number of tardy jobs and \(Max_{NT}\) is equal to 15% of the total number of jobs and presents a preference of the scheduler at Sherwood Press; there is full dissatisfaction when \(nTardy \geq Max_{NT}\), and full satisfaction when \(nTardy = 0\), while satisfaction degrees are linearly decreasing when \(nTardy\) increases from 0 to \(Max_{NT}\).

Parameter \(\lambda\) is introduced to define when job \(J_j\), \(j=1, \ldots, N\) is considered to be tardy. It is the case when \(S_{T}(\tilde{C}_j) \leq \lambda\), \(0 \leq \lambda \leq 1\). In our problem \(\lambda\) is set empirically to be 0.6.

(III) Satisfaction grade of Total Setup Time \(S_{ST}\) (Figure 8)

\[
S_{ST} = \begin{cases} 
\frac{1}{(Max_{ST} - Total_{ST})/Max_{ST}} & \text{if } Total_{ST} = 0 \\
0 & \text{if } 0 < Total_{ST} < Max_{ST} \\
0 & \text{if } Total_{ST} > Max_{ST}
\end{cases}
\tag{12}
\]

where \(Total_{ST} = \sum_{i=1}^{H} \sum_{j=1}^{F} aX_{ij}\), where a setup \(X_{ij}\) is applied whenever two jobs of different families are scheduled contiguously. We assume that a setup requires \(a = 20\) minutes each time it is applied.
A maximum setup time $\text{Max}_{ST}$ is needed when all consecutive jobs belong to different families, i.e., $\text{Max}_{ST} = \alpha \sum_{i=1}^{M} \sum_{j=1}^{N} (i, j)$, where $\sum_{i=1}^{M} \sum_{j=1}^{N} (i, j)$ is the total number of operations.

(IV) Satisfaction Grade of Total Idle times $S_{IT}$ (Figure 9)

$$S_{IT} = \begin{cases} 
1 & \text{if } \text{Total}_{IT} = 0 \\
(\text{Max}_{IT} - \text{Total}_{IT})/\text{Max}_{IT} & \text{if } 0 < \text{Total}_{IT} < \text{Max}_{IT} \\
0 & \text{if } \text{Total}_{IT} > \text{Max}_{IT}
\end{cases}$$

The Total Idle Time is

$$\text{Total}_{IT} = \sum_{i=1}^{M} \left( C_{ia} - s_{ab} \right) - \sum_{j=1}^{N} p_{ij} - H \sum_{i=1}^{M} \sum_{f=1}^{F} aX_{if} \beta$$

where $(i,a)$ and $(i,b)$ are the last operation finished and the first operation started on machine $M_i$ within planning horizon $H$, respectively.

$\sum_{j=1}^{N} p_{ij}$ is the total processing times of all jobs on machine $M_i$.

$\text{Max}_{IT}$ is the total time of machines being in use $\text{Max}_{IT} = \sum_{i=1}^{M} (C_{ia} - s_{ab})$.

(V) Satisfaction Grade of Total Throughput Time $S_{TT}$

$$S_{TT} = \begin{cases} 
1 & \text{if } \text{Total}_{TT} = 0 \\
(\text{Max}_{TT} - \text{Total}_{TT})/\text{Max}_{TT} & \text{if } 0 < \text{Total}_{TT} < \text{Max}_{TT} \\
0 & \text{if } \text{Total}_{TT} > \text{Max}_{TT}
\end{cases}$$

where the Total Throughput Time shows the time the jobs are on the shop floor, $\text{Total}_{TT} = \sum_{j=1}^{N} (C_{j} - s_{j})$.

Max Throughput Time is $\text{Max}_{TT} = \sum_{j=1}^{N} (C_{\text{max}} - s_{j})$.

Once the satisfaction grades with the values of all the objectives are determined, the average aggregation operator is used to combine them into one satisfaction grade $F$:

$$F = (S_{AT} + S_{NT} + S_{ST} + S_{IT} + S_{TT})/5$$

5. PERFORMANCE OF THE FUZZY GA ON THE REAL-WORLD DATA

Data obtained from Sherwood Press were used to test the performance of the developed fuzzy GA with lot-sizing and batching. The input data for each job are JobID, the order of machines that the job requires its processing on, uncertain processing time on each of the machines, required quantity of the items, family of the job, release date, due date and the priority of the job. The planning horizon is 4 weeks. Based on the release date and due date of the job, each job is assigned its urgency. The workload is calculated taking into consideration the required production of items. Jobs of the same family are grouped together into batches whenever possible. Batching is performed only on the printing machines.
Two groups of tests were performed with the aim (a) to evaluate the performance of the IF-THEN fuzzy rules for lot-sizing and (b) to evaluate the performance of the fuzzy GA which takes as input data the number of lots for each job (none or 2) and the size of each lot if the jobs are split into lots. The algorithm was implemented using Visual C++ in a Windows XP environment, and tests were run on a PC Pentium 2 GHz with 512 MB of RAM.

5.1 Testing of fuzzy rules for lot-sizing

A number of jobs with specified size and urgency, given in Table 1, were released. The workload of the shop floor was 5120125 items which required processing. The data in column change of lot shows how much the initial lot size of 50% has to be changed to reflect the size and the urgency of jobs. For example, job #7 of size 460, considered as Very Small (with membership degree 1) and of urgency 5, considered as High (with membership degree 0.25), will not be split into lots and therefore the total size of the first lot is 100% of the job. Also, two jobs (#1 and #2) of similar urgency will be split up differently due to a difference in their sizes. Job of the larger size (#1) will be split into lots of 37% and 63% of the size of the job, while in the case of job #2 of the smaller size, the larger part of the job (63%) will be processed in the first lot and the remaining part (37%) in the second lot. Considering jobs #3 and #6, which are of the same urgency, job #3 is larger and therefore its first lot is smaller (47%) than the second lot (53%). Job #6 is of the smaller size and the major part will be processed in the first lot (61%).

5.2 Testing of fuzzy genetic algorithm

The developed fuzzy GA was run with the parameters given in Table 2. The algorithm was run 5 times and the results obtained using real-world data for one month planning horizon discretised homogeneously into one hour unit time periods, are given in Table 3. This specific month was chosen because it was considered to be rather busy with 158 jobs and a workload of 5120125 required items. In order to study the effects of lot-sizing on the performance of the schedule, the algorithm was also run with no lot-sizing.

In the calculations of due-date oriented objective values achieved (AT and NT) in the schedules with lots we take into consideration the completion of jobs in the second lot (if the job was split in two lots). Therefore results presented in Table 3 need to be interpreted carefully having in mind that the splitting of jobs into lots leads to higher number of operations but gives better opportunities for batching. It can be observed that the average tardiness achieved in both cases, with and without lots, have similar values, while the number of tardy jobs is slightly lower in the schedules with lots (i.e., the average and the best satisfaction grades are higher). However, this is a consequence of the approach taken to calculate the tardiness of the schedule with lots when the second lot is taken into consideration only. In order to investigate the effect of splitting jobs into lots on tardiness, we also record the number of tardy jobs taking into consideration completion of jobs in the first lots. Table 4 shows the number of tardy jobs (out of 158) taking into consideration two lots and the first lot, respectively. Of course, the second lots determine the ultimate quality of the schedule but the smaller number of tardy jobs in the first lots may increase the satisfaction grade of the customer.

Contrary to the expectation, the obtained values of the total setup time are lower in the schedules with lots, although these schedules consist of higher number of operations. The possible explanation for that lies in batching. Namely, in schedules with lots, jobs are split up into operations of smaller sizes in terms of the number of items to be processed, i.e., there is a higher number of operations to be scheduled than in the schedules without lots. It leads to the smaller number of batches and higher average size of batches in the schedules with lots than in the schedules without lots. The total number of batches, the total number of operations to be scheduled and average size of batches on the 3 printing machines are given in Table 5 (there are 3 printing machines in Sherwood that batching can take place on).

The developed GA uses 6 different dispatching rules to sequence job operations on each of the machines. These dispatching rules are first chosen with the same probability. Further experiments are performed with assigning a higher probability to some of the rules. For example, in order to increase the satisfaction grade.
for the total setup time, the probability of applying the dispatching rule *Same Family of Jobs Together* is increased. As expected, the total setup time was improved (the satisfaction grade was increased from 0.327 to 0.6, but at an expense of the number of tardy jobs which increases from 11 to 73 (out of a total of 158 jobs). Similar experiments can be performed with other dispatching rules.

In the initial algorithm, the maximum number of lots was set to be 2. The question arises whether three lots can improve the performance of a schedule. The algorithm was applied as follows. The fuzzy IF-THEN rules are firstly applied to determine whether to split the job into lots and to decide on the size of the two lots. Then, the second lot was considered as a new problem and the same set of fuzzy rules was used to decide whether to split it into lots and to determine the sizes of the two additional lots. The obtained values of the objective functions are given in Table 6. We can conclude that they are similar to the values obtained with 2 lots.

6. CONCLUSION

The paper considers a real-world job shop problem that is present in a printing company. Processing times of jobs and their due dates are imprecise and modelled by fuzzy sets. Fuzzy rules, which reflect the scheduling policy of the printing company, are defined to determine lot sizes. A fuzzy multi-objective genetic algorithm is developed to generate schedules of jobs. The objectives considered are to minimize average tardiness, number of tardy jobs, setup times, idle times of machines and throughput times of jobs. Fuzzy sets are used to represent satisfaction grades for the objectives taking into consideration the preferences of the scheduler on the shop floor. A genetic algorithm is developed to search for a schedule with maximum satisfaction grades for the objectives. The algorithm uses lot sizing of jobs determined by the fuzzy rule-based system. In addition, the batching of jobs with similar characteristics is taken into consideration, which leads to reduce the total setup time of machines. The results obtained on real-world data from a printing company are given and analyzed.

Our future work will be focused on additional issues regarding the lot sizing and batching. We will consider scheduling of lots on the alternative parallel machines on the shop floor at the same time. Batching of jobs is important in real-world problems because it reduces setup time of the machines. In the current algorithm it is handled by one dispatching rule, which groups jobs of the same family together in one batch. We will investigate whether a new objective, which maximises number of batches can improve the performance of the schedule.

ACKNOWLEDGMENTS

The authors would like to thank the Engineering and Physics Science Research Council (EPSRC), UK, for supporting this research (Grant No. GR/R95319/01). We also acknowledge the support of the industrial collaborator Sherwood Press Ltd, Nottingham.

REFERENCES


APPENDIX

Fuzzy set $\tilde{A}$ is defined by a membership function $\mu_A(x)$ which assigns to each object $x$ in the universe of discourse $X$, a value that represents its grade of membership in the fuzzy set (Klir, 1998):

$$\mu_A(x) : X \rightarrow [0,1]$$

A variety of shapes can be used for membership functions such as triangular, trapezoidal, bell curves and $s$-curves. Conventionally, the choice of the shape is subjective and allows the decision maker to express his/her preferences.
Table 1. Change of lot obtained using the fuzzy IF-THEN rules

<table>
<thead>
<tr>
<th>JobID</th>
<th>size</th>
<th>urgency</th>
<th>change of lot</th>
<th>First lot</th>
<th>Second lot</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>70000</td>
<td>19</td>
<td>-13%</td>
<td>37%</td>
<td>63%</td>
</tr>
<tr>
<td>#2</td>
<td>4180</td>
<td>18</td>
<td>+13%</td>
<td>63%</td>
<td>37%</td>
</tr>
<tr>
<td>#3</td>
<td>70000</td>
<td>10</td>
<td>-3%</td>
<td>47%</td>
<td>53%</td>
</tr>
<tr>
<td>#4</td>
<td>4500</td>
<td>3</td>
<td>+29%</td>
<td>79%</td>
<td>21%</td>
</tr>
<tr>
<td>#5</td>
<td>596000</td>
<td>8</td>
<td>-12%</td>
<td>38%</td>
<td>62%</td>
</tr>
<tr>
<td>#6</td>
<td>12000</td>
<td>10</td>
<td>+11%</td>
<td>61%</td>
<td>39%</td>
</tr>
<tr>
<td>#7</td>
<td>460</td>
<td>5</td>
<td>+50%</td>
<td>100%</td>
<td>0%</td>
</tr>
<tr>
<td>#8</td>
<td>84000</td>
<td>35</td>
<td>-25%</td>
<td>25%</td>
<td>75%</td>
</tr>
<tr>
<td>#9</td>
<td>140000</td>
<td>12</td>
<td>-15%</td>
<td>35%</td>
<td>65%</td>
</tr>
<tr>
<td>#10</td>
<td>60000</td>
<td>12</td>
<td>-3%</td>
<td>47%</td>
<td>53%</td>
</tr>
</tbody>
</table>

Table 2. Parameters of the GA

<table>
<thead>
<tr>
<th>Population size</th>
<th>Length of the chromosome</th>
<th>Crossover probability</th>
<th>Mutation probability</th>
<th>Termination condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>$2M$, ($M$ is the number of machines)</td>
<td>0.8</td>
<td>0.3</td>
<td>500 iterations</td>
</tr>
</tbody>
</table>

Table 3. Average and best values of the aggregated satisfaction grades and satisfaction grades of the objectives, runs with and without lots

<table>
<thead>
<tr>
<th>Runs with lots</th>
<th>$F$</th>
<th>$S_{st}$</th>
<th>$S_{ct}$</th>
<th>$S_{st}$</th>
<th>$S_{st}$</th>
<th>$S_{ct}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.51</td>
<td>0.424</td>
<td>0.588</td>
<td>0.345</td>
<td>0.368</td>
<td>0.843</td>
</tr>
<tr>
<td>2</td>
<td>0.54</td>
<td>0.43</td>
<td>0.651</td>
<td>0.371</td>
<td>0.372</td>
<td>0.874</td>
</tr>
<tr>
<td>3</td>
<td>0.53</td>
<td>0.43</td>
<td>0.652</td>
<td>0.353</td>
<td>0.369</td>
<td>0.873</td>
</tr>
<tr>
<td>4</td>
<td>0.53</td>
<td>0.43</td>
<td>0.652</td>
<td>0.336</td>
<td>0.368</td>
<td>0.877</td>
</tr>
<tr>
<td>5</td>
<td>0.52</td>
<td>0.43</td>
<td>0.683</td>
<td>0.239</td>
<td>0.373</td>
<td>0.859</td>
</tr>
<tr>
<td>Average</td>
<td>0.53</td>
<td>0.43</td>
<td>0.645</td>
<td>0.329</td>
<td>0.37</td>
<td>0.865</td>
</tr>
<tr>
<td>Best</td>
<td>0.54</td>
<td>0.43</td>
<td>0.652</td>
<td>0.363</td>
<td>0.37</td>
<td>0.87</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Runs without lots</th>
<th>$F$</th>
<th>$S_{st}$</th>
<th>$S_{ct}$</th>
<th>$S_{st}$</th>
<th>$S_{st}$</th>
<th>$S_{ct}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.46</td>
<td>0.416</td>
<td>0.525</td>
<td>0.119</td>
<td>0.384</td>
<td>0.866</td>
</tr>
<tr>
<td>2</td>
<td>0.47</td>
<td>0.429</td>
<td>0.588</td>
<td>0.1</td>
<td>0.379</td>
<td>0.874</td>
</tr>
<tr>
<td>3</td>
<td>0.46</td>
<td>0.42</td>
<td>0.556</td>
<td>0.11</td>
<td>0.38</td>
<td>0.86</td>
</tr>
<tr>
<td>4</td>
<td>0.39</td>
<td>0.419</td>
<td>0.557</td>
<td>0.113</td>
<td>0.379</td>
<td>0.866</td>
</tr>
<tr>
<td>5</td>
<td>0.42</td>
<td>0.407</td>
<td>0.367</td>
<td>0.11</td>
<td>0.375</td>
<td>0.847</td>
</tr>
<tr>
<td>Average</td>
<td>0.44</td>
<td>0.418</td>
<td>0.518</td>
<td>0.110</td>
<td>0.379</td>
<td>0.862</td>
</tr>
<tr>
<td>Best</td>
<td>0.47</td>
<td>0.429</td>
<td>0.588</td>
<td>0.119</td>
<td>0.384</td>
<td>0.874</td>
</tr>
</tbody>
</table>
### Table 4. Number of tardy jobs versus tardy first lots

<table>
<thead>
<tr>
<th>Instance</th>
<th>Tardy jobs</th>
<th>Tardy first lots</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>14</td>
<td>8</td>
</tr>
</tbody>
</table>

### Table 5. Batching with and without lots

<table>
<thead>
<tr>
<th>Run with</th>
<th>Number of batches</th>
<th>Number of operations</th>
<th>Batch Average Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>no lots</td>
<td>162</td>
<td>185</td>
<td>1.13</td>
</tr>
<tr>
<td>lots</td>
<td>179</td>
<td>369</td>
<td>1.83</td>
</tr>
</tbody>
</table>

### Table 6. Average and best values of the aggregated satisfaction grades and satisfaction grades of the objectives with 3 lots

<table>
<thead>
<tr>
<th>Run with 3 lots</th>
<th>$F$</th>
<th>$S_{ST}$</th>
<th>$S_{IT}$</th>
<th>$S_{IT}$</th>
<th>$S_{TT}$</th>
<th>$S_{TT}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.55</td>
<td>0.43</td>
<td>0.65</td>
<td>0.44</td>
<td>0.38</td>
<td>0.87</td>
</tr>
<tr>
<td>2</td>
<td>0.54</td>
<td>0.43</td>
<td>0.65</td>
<td>0.42</td>
<td>0.39</td>
<td>0.87</td>
</tr>
<tr>
<td>3</td>
<td>0.52</td>
<td>0.43</td>
<td>0.65</td>
<td>0.28</td>
<td>0.39</td>
<td>0.86</td>
</tr>
<tr>
<td>4</td>
<td>0.52</td>
<td>0.43</td>
<td>0.65</td>
<td>0.28</td>
<td>0.38</td>
<td>0.86</td>
</tr>
<tr>
<td>5</td>
<td>0.50</td>
<td>0.42</td>
<td>0.55</td>
<td>0.34</td>
<td>0.38</td>
<td>0.85</td>
</tr>
<tr>
<td>Average</td>
<td>0.53</td>
<td>0.428</td>
<td>0.63</td>
<td>0.352</td>
<td>0.384</td>
<td>0.862</td>
</tr>
<tr>
<td>Best</td>
<td>0.55</td>
<td>0.43</td>
<td>0.65</td>
<td>0.44</td>
<td>0.38</td>
<td>0.87</td>
</tr>
</tbody>
</table>
Figure 1. Fuzzy processing times

Figure 2. Fuzzy due dates
Job Shop Scheduling with Lot-Sizing and Batching in an Uncertain Real-World Environment

Figure 3. Linguistic terms for variables workload, urgency of the job, size of the job
Figure 4. Membership function of the linguistic terms of the conclusion variable

Figure 5. Membership function of the linguistic terms of the conclusion variable

1. IF workload is Not Large THEN change of lot is Small
2. IF workload is Large THEN change of lot is Medium Negative
3. IF size of job is Very Small THEN change of lot is Large Positive
4. IF size of job is Small THEN change of lot is Medium Positive
5. IF size of job is Medium THEN change of lot is Medium Negative
6. IF size of job is Large THEN change of lot is Large Negative
7. IF urgency of job is High THEN change of lot is Large Positive
8. IF urgency of job is Medium THEN change of lot is Medium Positive
9. IF urgency of job is Low THEN change of lot is Large Negative

Figure 5. Fuzzy rules for deriving the size of the lots
Figure 6. Satisfaction grade of completion time using area of intersection

Figure 7. Satisfaction grade of Number of Tardy Jobs

Figure 8. Satisfaction grade of Total Setup Time

Figure 9. Satisfaction grade of Total Idle Time

Figure 10. Satisfaction grade of Total Throughput Time