

AN INTEGRATED APPROACH FOR SINGLE MACHINE SCHEDULING WITH SEQUENCE-DEPENDENT SETUP TIMES

Derya Deliktas*

Department of Industrial Engineering
Dumlupınar University
Kütahya, Turkey
E-mail: deryadeliktas@hotmail.com

Orhan Torkul

Department of Industrial Engineering
Sakarya University
Sakarya, Turkey
E-mail: torkul@sakarya.edu.tr

Ozden Ustun

Department of Industrial Engineering
Dumlupınar University
Kütahya, Turkey
E-mail: ozden.ustun@dpu.edu.tr

Safak Kiris

Department of Industrial Engineering
Dumlupınar University
Kütahya, Turkey
E-mail: safak.kiris@dpu.edu.tr

ABSTRACT

This study proposes a multi-choice goal programming for the single machine scheduling problem of minimizing the weighted number of tardy jobs, the total weighted completion time and makespan with sequence-dependent setup times. In this problem, there are n candidate jobs for processing in a single machine, each job has a weight, a due date, a processing time, and also sequence-dependent setup times exist between two consecutive jobs. In the first stage of the proposed methodology, the job weights of each job are determined by using the Analytic Hierarchy Process (AHP) method. In the second stage, a 0-1 mixed integer non-linear programming model is built by considering three objective functions and the ideal point is obtained by minimizing the objectives individually. Then, the multi-choice goal programming is used to allow the decision makers to set multi-choice aspiration levels for each goal.

Keywords: AHP; single machine scheduling problem; sequence-dependent setup times; Multi-choice goal programming; 0-1 mixed integer non-linear programming model

1. Introduction

In the field of scheduling problems sequence-dependent setup times and job weights are very important elements, and non-execution of proper scheduling and sequencing of jobs will cause a significant increase in both makespan and the number of tardy jobs (Bahalke et al., 2010). In classical scheduling problems, it is reasonable and necessary to consider scheduling problems with setup times and job weights. Setup includes work to prepare the machine, process, or bench for product parts or the cycle. This includes obtaining tools, positioning work-in-process material, return tooling, cleanup, setting the required jigs and fixtures, adjusting tools, and inspecting material. Scheduling problems involving setup times can be divided into two classes; the first class is sequence-independent and the second is sequence-dependent setup times. Setup is sequence-dependent if its duration depends on both the current and the immediately preceding job, and is sequence-independent if its duration depends only on the current job to be processed (Allahverdi et al., 1999).

Sequence-dependent setup times are usually found in situations where the facility has a multipurpose machine. Some examples of sequence-dependent setups include (i) chemical compounds manufacturing, where the extent of the cleansing depends on both the chemical most recently processed and the chemical about to be processed, and (ii) the printing industry, where the cleaning and setting of the presses for processing the next job depends on its difference from the colour of ink, size of paper and types used in the previous job. The case of sequence-dependent setups can be found in numerous other industrial systems, which include the stamping operation in plastic manufacturing, die changing a metal processing shop, and roll slitting in the paper industry (Eren & Güner, 2006).

In the literature, there are numerous studies that focus on single machine scheduling problems with sequence-dependent setup times. Panwalker and Iskander (1977) dealt with a scheduling problem which involved both the completion time and the tardiness as criteria and sequence-dependent setup times on single machine case. Bianco et al. (1993), Fischetti et al. (1993), Arcelus and Chandra (1983) and Miyazaki and Ohta (1987) developed exact solution methods for minimizing total completion time problems with sequence-dependent setup. Tan and Narasimhan (1997) developed a simulated annealing algorithm for the problem of minimizing tardiness, a common measure of due-date performance, in a sequence-dependent setup environment. Wang and Wang (1997) formed a hybrid algorithm by combining the heuristic with a genetic algorithm for single machine earliness-tardiness scheduling problems with sequence-dependent setup time under a different due date. Asano and Ohta (1999) developed an optimization algorithm based on the branch-and-bound method to minimize the maximum tardiness. Armentano and Mazzini (2000) presented a genetic algorithm for a single machine scheduling problem with the objective of minimizing total tardiness. Tan et al. (2000) considered the problem of scheduling a single machine for minimizing total tardiness in a sequence-dependent setup environment with the comparative performance of branch-and-bound, genetic search, simulated annealing and random-start pairwise interchange. Franca et al. (2001) proposed a new memetic algorithm for the total tardiness single machine scheduling problem with due dates and sequence-dependent setup times. Gagne et al. (2002) developed ant colony optimization algorithm to minimize total tardiness. Mendes et al. (2002) proposed a memetic algorithm and a multiple start approach for the solution of the single machine scheduling problem with sequence-dependent setup times. Shin et

al. (2002) presented a tabu search algorithm in order to minimize the maximum lateness of the jobs. Chang et al. (2004) attempted to solve a single machine scheduling problem in which the objective function was to minimize the total weighted tardiness. Lee and Asllani (2004) developed an integer programming model and a genetic algorithm for a bi-criteria problem (objective of minimizing makespan and number of tardy jobs) with sequence-dependent setup times. Rabadi et al. (2004) developed a branch-and-bound algorithm to minimize the total amount of earliness and tardiness. Eren and Güner (2006) proposed an integer programming model for minimization of the weighted sum of total completion time and total tardiness. Gupta and Smith (2006) proposed two algorithms, a problem space-based local search heuristic and a Greedy Randomized Adaptive Search Procedure to minimize total tardiness with sequence dependent setup times. Kırış and Saraç (2009) presented fuzzy goal programming for scheduling the problem of single machine with sequence dependent setup times. Bahalke et al. (2010) developed genetic and tabu search algorithms for minimizing makespan with sequence-dependent setup times on a single machine case. Sioud et al. (2012) presented a hybrid approach based on the integration between a genetic algorithm and concepts from constraint programming, multi-objective evolutionary algorithms and ant colony optimization for minimizing the total tardiness. There are also studies related to parallel machine scheduling with sequence-dependent setup times. Chen (2012) examined the unrelated parallel machine scheduling problem with unequal ready times and sequence-dependent setup times to minimize the weighted number of tardy jobs. Chen (2013) also dealt with the unrelated parallel machine scheduling to minimize total weighted completion time with sequence-dependent setup times.

In this study, we are seeking to minimize the weighted number of tardy jobs, the total weighted completion time and makespan on a single machine under the existence of sequence-dependent setup times. To the best of our knowledge research into this problem using AHP and multi-choice goal programming has not been done. Although many researchers focus on the tangible criteria, intangible criteria cannot be neglected in real life problems, and the AHP considers both tangible and intangible criteria. A few studies looking at scheduling problems using the AHP method do exist. Wu et al. (2007) used the AHP method to select the best solution for the multi-objective flexible job shop scheduling problem. Witkowski et al. (2009) presented an evaluation of a job shop scheduling problem under multiple objectives by using the AHP method for comparing schedules in accordance with multiple objectives. Di and Ze (2011) proposed a hybrid genetic-tabu search algorithm to solve the flexible job-shop scheduling problem and adopted an AHP application to translate a multi-objective problem into single objective one. Fang et al. (2011) combined AHP with grey relational analysis to help make the decision about pre-processing equipment under multi-objective conditions for a multi-objective job shop scheduling problem. Lin et al. (2012) used the AHP method to make the parent selection in a genetic algorithm to solve a hybrid flow shop scheduling problem. Goal programming is an analytical approach devised to address decision making problems where targets have been assigned to all the attributes and where the decision maker is interested in minimizing the non-achievement of the corresponding goals (Romero, 2004). Chang (2007) has recently proposed a novel approach namely multi-choice goal programming, which allows decision makers to set multi-choice aspiration levels for each goal (i.e., one goal mapping multiple aspiration levels) to avoid underestimation of decision making. According to multi-choice goal programming, decision makers must not only consider the single aspiration level in the local region, but

also develop multiple aspiration levels under given constraints to obtain the global optimal solution in the global region. Chang (2008) proposed an alternative method to formulate multi-choice goal programming in which the new approach didn't involve multiplicative terms of binary variables for solving such problems. An efficient multi-choice goal programming formulation based on the conic scalarizing function is proposed by Ustun (2012) with three contributions: (1) the alternative formulation allows the decision maker to set multi-choice aspiration levels for each goal to obtain an efficient solution in the global region, (2) the proposed formulation reduces auxiliary constraints and additional variables, and (3) the proposed model guarantees to obtain a properly efficient (in the sense of Benson) point. The rapid development of multi-choice goal programming has led to an enormous diversity in models and applications. In practice, the multi-choice goal programming has been applied to the real-world multi-criteria decision making problems, such as supplier selection (Liao and Kao, 2010; Paksoy and Chang, 2010), an evaluation of framework for product planning (Lee et al., 2010), the plotting a quality management system (Ben Mahmoud et al., 2010), the aggregate production planning (Da Silva et al., 2013a, 2013b). According to the standard classification of scheduling problems for three-field notation provided by Pinedo (1995), this problem is denoted by $1|s_{ij}|\sum w_j U(n), \sum w_j C(n), C_{max}$.

This paper presents a methodology to solve single machine scheduling problems with sequence-dependent setup times in Section 2. In the proposed methodology, multi-objective a programming model is combined with the AHP method to determine the job weights. Additionally, Section 2 deals with an example which integrates the proposed 0-1 mixed integer non-linear programming model formulations and then multi-choice goal programming formulation to find the satisfactory job sequence. Conclusions are given in Section 3.

2. Materials and methodology

In this study, a single machine scheduling problem of minimizing the weighted number of tardy jobs, the total weighted completion time and makespan with sequence-dependent setup times was considered. Firstly, a 0-1 mixed integer non-linear programming model was provided. There were 6 jobs with 3 customers, and the problem was assumed to be without preemption and breakdown. All jobs were ready at time zero. Processing times, due dates and initial setup times are given in Table 1. Also, the sequence-dependent setup times that exist between the job pairs are given in Table 2.

Table 1
Parameters of the jobs

Jobs	1	2	3	4	5	6
Processing time (p_j)	12	8	3	10	4	18
Due date (d_j)	10	2	72	11	24	60
Initial Setup time	5	4	6	7	10	3

Table 2
Setup time of switching from i to job j

Jobs	1	2	3	4	5	6
1	---	7	8	6	14	15
2	5	---	18	20	5	8
3	3	11	---	19	9	10
4	7	12	1	---	6	11
5	8	4	8	3	---	16
6	9	2	7	1	2	---

A three-stage solution methodology was proposed to solve the multi-objective scheduling problem. The flow chart of the proposed methodology is given in Figure 1. In the first stage, the jobs were evaluated by the AHP to obtain the job weights. These weights were used as parameters for the objectives of the weighted number of tardy jobs and the total weighted completion time in the second stage. Also, the ideal point was determined to define the multi-choice goals. The multi-choice goal programming model was constructed and solved to obtain a satisfactory schedule in stage 3.

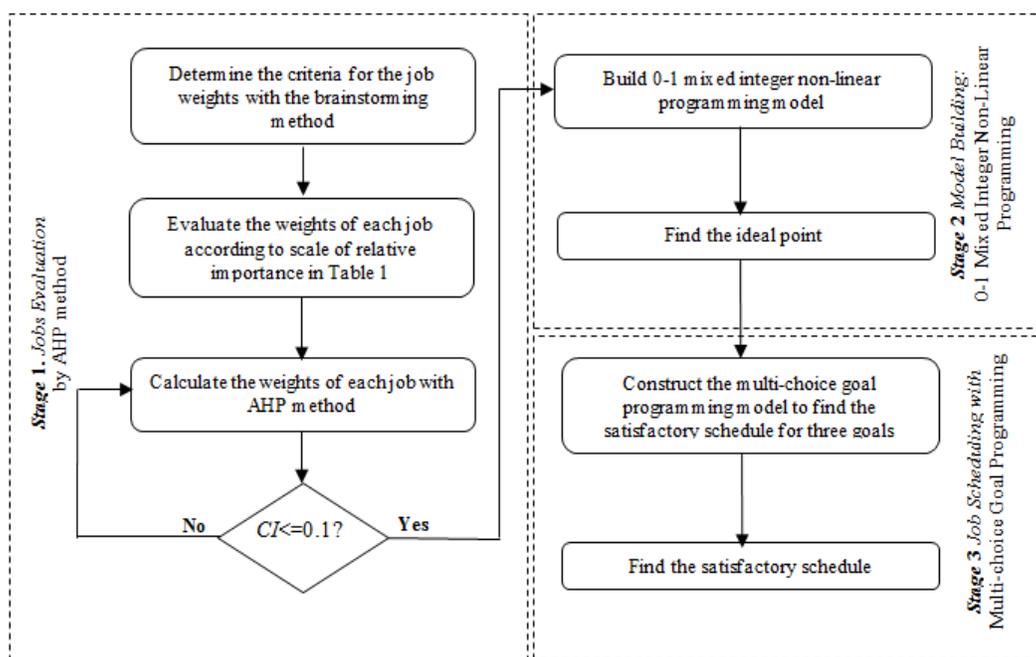


Figure 1. Flow chart of proposed methodology

2.1 Stage 1: Job evaluation with AHP method

The AHP, developed by Saaty (1980), is a technique which considers data or information for a decision in a systematic manner. AHP is mainly concerned with solving decision problems with uncertainties in a multiple criteria characterization. It is based on three principles: (1) constructing the hierarchy, (2) priority setting, and (3) logical consistency (Fazlollahtabar et al., 2011).

2.1.1 Construction of the hierarchy

A multi-criteria decision making problem is structured and decomposed into sub-problems (sub-objectives, criteria, alternatives, etc.), within the hierarchy. The objective level on the top is the determination of weights of each job, and the second level is the criteria used to determine the weight of each job, including customers and suppliers. The third level is the sub-criteria of each criterion, including reliability, orders frequency / size, work period, net profit, prestige, cost, quality, inventory level and on time delivery, and the bottom level is the six alternatives dealt with in this study. The hierarchy of the proposed problem is given in Figure 2.

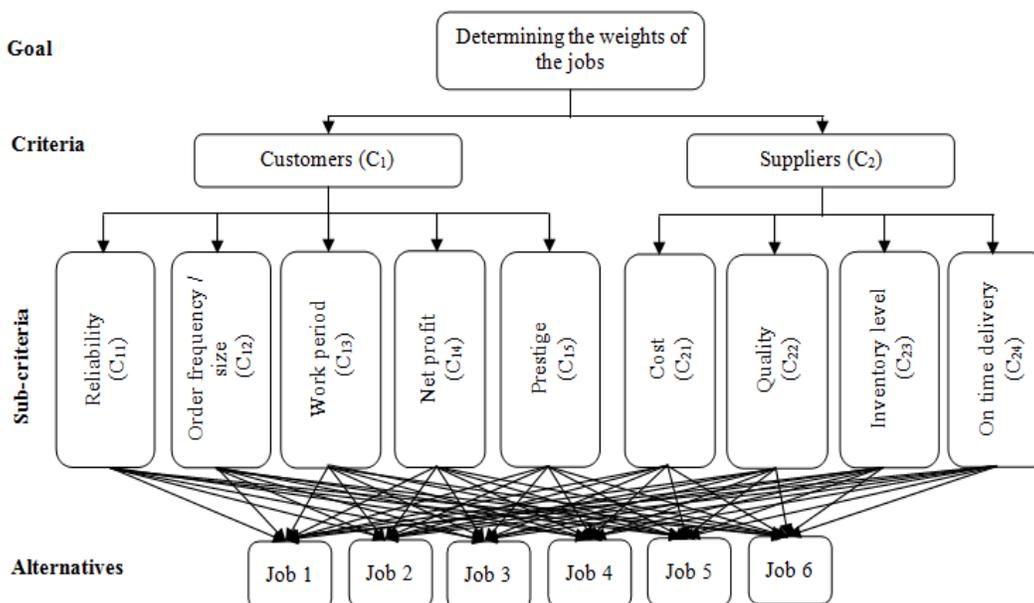


Figure 2. Hierarchy of proposed problem

2.1.2 Priority setting

The relative “priority” given to each element in the hierarchy was determined by pairwise comparison of the contributions of elements at a lower level in terms of the criteria (or elements) with a causal relationship (Macharis et al., 2004). The decision maker uses a pairwise comparison mechanism, shown in Table 3 of Saaty (2000) and a 1–9 scale.

Let $C = \{C_j | j = 1, 2, \dots, m\}$ be the set of criteria. The result of the pair-wise comparison on m criteria can be summarized in an $m \times m$ evaluation matrix A in which every element a_{ij} is the quotient of weights of the criteria, as shown in (1) below:

$$A = (a_{ij}), \quad i, j = 1, \dots, m \tag{1}$$

The relative priorities are given by the right eigenvector (θ) corresponding to the largest eigenvector (λ_{max}) as:

$$A\theta = \lambda_{max}\theta \tag{2}$$

In case the pair-wise comparisons are completely consistent, the matrix A has rank 1, and $\lambda_{max} = m$. In that case, weights can be obtained by normalizing any of the rows or columns of A .

The procedure described above is repeated for all subsystems in the hierarchy. In order to synthesize the various priority vectors, these vectors are weighted with the global priority of the parent criteria and synthesized starting at the top of the hierarchy. As a result, the overall relative priorities to be given to the lowest level elements are obtained. These overall, relative priorities indicate the degree to which the alternatives contribute to the focus. These priorities represent a synthesis of the local priorities, and reflect an evaluation process that permits integration of the perspectives of the various stakeholders involved (Fazlollahtabar et al., 2011).

2.1.3 Consistency check

A measure of consistency of the given pair-wise comparison is needed. The consistency is defined by the relation between the entries of A ; that is, we say A is consistent if $a_{ij} \cdot a_{jk} = a_{ik}$, for each i, j, k . If the pair-wise comparisons do not include any inconsistencies, then $\lambda_{max} = m$. The more consistent the comparisons are, the closer the value of computed λ_{max} is to m . A consistency index (CI), which measures the inconsistencies of pair-wise comparisons, is set to be:

$$CI = \frac{(\lambda_{max} - m)}{(m - 1)} \quad (3)$$

The final consistency ratio (CR), on the basis of which one can conclude whether the evaluations are sufficiently consistent, is calculated as the ratio of the CI and the random consistency index (RI), as indicated in Equation 4 below:

$$CR = 100 \left(\frac{CI}{RI} \right) \quad (4)$$

where m is the number of columns in A and RI is the random index given in Table 4, being the average of the CI obtained from a large number of randomly generated matrices. Note that RI depends on the order of the matrix, and a CR value of 10% or less is considered acceptable (Saaty, 1980). The value 0.1 is the accepted upper limit for CR . If the final consistency ratio exceeds this value, the evaluation procedure needs to be repeated to improve consistency. The measurement of consistency can be used to evaluate the consistency of decision makers as well as the consistency of all the hierarchies.

Table 3
Scale of relative importance

Intensity scale of importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to the objective
2	Weak	
3	Moderate importance	Experience and judgment slightly favor one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgment strongly favor one activity over another
6	Strong plus	
7	Very strong or demonstrated Importance	An activity is favored very strongly over another; its dominance demonstrated in practice
8	Very, very strong	
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation
Reciprocals of above	If activity i has one of the above nonzero numbers assigned to it when compared with activity j , then j has the reciprocals value when compared with i	A reasonable assumption
Rationals	Ratios arising from the scale	If consistency were to be forced by obtaining n numerical values to span the matrix

Table 4
The random consistency index

r	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RI	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.48	1.56	1.57	1.59

Three experts from the purchasing, production and marketing departments made the pair-wise comparison for each criteria and sub-criteria. For example, the pair-wise comparisons matrices for main criteria (C_1 and C_2), the customers' sub-criteria (C_{11} , C_{12} , C_{13} , C_{14} , C_{15}) and jobs for customer reliability sub-criterion are given in Table 5.

The final weights of the jobs are obtained by using AHP as $w = (0.2182, 0.2163, 0.2200, 0.1450, 0.1022, 0.0984)$.

Table 5
Pair-wise comparison matrix and weights from AHP for the evaluation hierarchy criteria and alternative level

Level or sub-level of evaluation hierarchy	Pair-wise comparison matrix					Local weight	Overall weight		
	C ₁	C ₂							
Criteria level	C ₁	1.00	3.00			0.7500			
	C ₂	0.33	1.00			0.2500			
When the dimension of pair-wise comparison matrix equals to 2, the CR needs not to be calculated.									
Level or sub-level of evaluation hierarchy	Pair-wise comparison matrix					Local weight	Overall weight		
	C ₁₁	C ₁₂	C ₁₃	C ₁₄	C ₁₅				
Sub-levels of customers criterion	C ₁₁	1.00	3.00	2.00	1.00	0.50	0.2171	0.1628	
	C ₁₂	0.33	1.00	2.00	0.33	0.33	0.1111	0.0833	
	C ₁₃	0.50	0.50	1.00	0.50	0.33	0.0956	0.0717	
	C ₁₄	1.00	3.00	2.00	1.00	0.50	0.2176	0.1632	
	C ₁₅	2.00	3.00	3.00	2.00	1.00	0.3586	0.2690	
$\lambda_{max} = 5.1802; CI = 0.045; RI = 1.12; CR = 0.0402$									
Level or sub-level of evaluation hierarchy	Pair-wise comparison matrix						Local weight	Overall weight	
	J ₁	J ₂	J ₃	J ₄	J ₅	J ₆			
Jobs for reliability criterion	J ₁	1.00	1.00	1.00	3.00	3.00	2.00	0.2386	0.0388
	J ₂	1.00	1.00	1.00	3.00	3.00	2.00	0.2386	0.0388
	J ₃	1.00	1.00	1.00	3.00	3.00	2.00	0.2386	0.0388
	J ₄	0.33	0.33	0.33	1.00	1.00	0.50	0.0761	0.0124
	J ₅	0.33	0.33	0.33	1.00	1.00	0.50	0.0761	0.0124
	J ₆	0.50	0.50	0.50	2.00	2.00	1.00	0.1321	0.0215
$\lambda_{max} = 6.0179; CI = 0.0036; RI = 1.24; CR = 0.0029$									

2.2 Stage 2: Model Building: 0-1 Mixed Integer Non-Linear Programming

In this stage, a single machine scheduling problem of minimizing the weighted number of tardy jobs, the total weighted completion time and makespan with sequence-dependent setup times was analyzed. The problem was assumed to be without preemption and breakdown, and all jobs were ready at time zero. The problem was defined as a 0-1 mixed integer non-linear programming model with three objectives. The indices, the parameters and the variables used in 0-1 mixed integer non-linear programming model and multi-choice goal programming model are given in Tables 6, 7 and 8, respectively. Parameters of the jobs and setup times are given in Tables 1 and 2.

Table 6
The indices

Index	Definition
<i>I</i> / <i>J</i>	Job index used as a unique identifier for each job. $I / J = \{1, \dots, 6\}$
<i>K</i>	Job index used to identify the position of a job in a given sequence. $K = \{1, \dots, 6\}$

Table 7
The parameters

Parameter	Definition
p_j	Processing time for job j
w_j	Job weight for job j
d_j	Due time for job j
S_{0j}	Setup time of job j in the first sequence position (initial set up time)
S_{ij}	Incremental setup time of switching from job i to job j
M	A positive large number
n	Number of jobs ($n = 6$)

Table 8
The variables

Variable	Definition
x_{jk}	1 if job j is assigned to the k th position in the sequence; 0, otherwise
x_{ijk}	1 if job j is assigned to the k th position in the sequence and preceded by job i ; 0, otherwise
$U(k)$	1 if job in the k th position is tardy; 0, otherwise
$C(k)$	Completion time for the job in the k th position in the sequence
$S(k)$	Setup time for the job in the k th position in the sequence
$P(k)$	Processing time for the job in the k th position in the sequence
$d(k)$	Due time for the job in the k th position in the sequence

Using the above notations, a 0-1 mixed integer non-linear programming model formulation can be given as below:

$$\text{Min } Z_1 = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} U(k) \tag{6}$$

$$\text{Min } Z_2 = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} C(k) \tag{7}$$

$$\text{Min } Z_3 = C(n) \tag{8}$$

Subject to

$$\sum_{j=1}^n x_{jk} = 1, \quad k = 1, \dots, 6 \tag{9}$$

$$\sum_{k=1}^n x_{jk} = 1, \quad j = 1, \dots, 6 \tag{10}$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} = 1, \quad i \neq j \quad k = 2, \dots, 6 \tag{11}$$

$$x_{jk} + x_{ik-1} - 1 \leq x_{ijk}, \quad i = 1, \dots, 6; j = 1, \dots, 6; k = 2, \dots, 6 \text{ and } i \neq j \tag{12}$$

$$S(1) = \sum_{j=1}^n S_{0j} x_{j1}, \tag{13}$$

$$S(k) = \sum_{j=1}^n \sum_{i=1}^n S_{ij} x_{ijk}, \quad i \neq j \quad k = 2, \dots, 6 \tag{14}$$

$$P(k) = \sum_{j=1}^n x_{jk} p_j, \quad k = 1, \dots, 6 \tag{15}$$

$$C(1) = S(1) + P(1) \tag{16}$$

$$C(k) = C(k-1) + S(k) + P(k), \quad k = 2, \dots, 6 \tag{17}$$

$$d(k) = \sum_{j=1}^n x_{jk} d_j, \quad k = 1, \dots, 6 \tag{18}$$

$$-C(k) + d(k) \leq M(1 - U(k)), \quad k = 1, \dots, 6 \tag{19}$$

$$C(k) - d(k) \leq MU(k), \quad k = 1, \dots, 6 \tag{20}$$

$$U(k), x_{jk}, x_{ijk} : 0\text{-}1 \text{ integer}, \quad i = 1, \dots, 6; j = 1, \dots, 6 \text{ and } k = 1, \dots, 6 \tag{21}$$

$$C(k), S(k), P(k), d(k) \geq 0, \quad k = 1, \dots, 6 \tag{22}$$

The objective function sets, Equations 6-8, are constructed to minimize the weighted number of tardy jobs, and the total weighted completion time and makespan, respectively. Equations 9 and 10 guarantee that only one job is assigned to each sequence position and only one sequence location, respectively. Equations 11 and 12 guarantee that only one job, job j , is assigned to follow job i . Equation 13 determines the setup time of the first sequence position, while Equation 14 determines the setup time of the k th ($k > 1$) sequence position. Equation 15 determines the processing time, and Equation 16 determines the completion time of the first sequence position. Equation 17 determines the completion time of the k th ($k > 1$) sequence position, and Equation 18 identifies the due date of the job in the k th sequence position. Equations 19 and 20 identify the tardy positions of the job sequence, and finally Equations 21 and 22 represent the integrality and non-negativity constraints.

2.3 Stage 3: Job scheduling with multi-choice goal programming

Multi-choice goal programming can be described in the following cases.

The first case: “the less the better” is formulated as:

$$\begin{aligned} & \text{Min } \sum_{i=1}^3 [(\beta + \alpha_i)d_i^+ + (\beta - \alpha_i)d_i^-], \\ & \text{subject to} \\ & f_i(x) - d_i^+ + d_i^- = y_i, i = 1,2,3, \\ & a_{i,min} \leq y_i \leq a_{i,max}, i = 1,2,3, \\ & d_i^+, d_i^- \geq 0, i = 1,2,3, \\ & x \in X \text{ (} X \text{ is a feasible set),} \end{aligned} \tag{23}$$

where the i th aspiration level y_i is the continuous variable restricted between the upper ($a_{i,max}$) bound and lower ($a_{i,min}$) bound ($a_{i,min} \leq y_i \leq a_{i,max}$); and d_i^+ and d_i^- are positive and negative deviations attached to the i th goal $|f_i(x) - y_i|$ in Equation 23; where $d_i^+ = \max(0, f_i(x) - a_i)$ and $d_i^- = \max(0, a_i - f_i(x))$ are, respectively, over and under achievements of i th goal; where $(\beta + \alpha_i)$ is the positive weight attached to positive deviation d_i^+ and $(\beta - \alpha_i)$ is the negative weight attached to negative deviation d_i^- for i th goal. The weight $(\beta + \alpha_i)$ is strictly positive and other weight $(\beta - \alpha_i)$ is strictly negative because of $(\beta, \alpha) \in W = \{(\beta, \alpha) \in R \times R_+^3 : 0 \leq \beta < \min\{\alpha_1, \alpha_2, \alpha_3\}\}$. If over-achievement is considered more desirable than under-achievement then β should be selected close to α_i as soon as possible by considering $(\beta, \alpha) \in W$.

In the second case: “the more the better”, objective functions which are maximized in the model can be easily transformed to “the less the better” form by multiplying -1 (Ustun, 2012).

The weights of objectives of minimizing the weighted number of tardy jobs, the total weighted completion time and makespan are used as 0.4, 0.3 and 0.3, respectively. Then each objective function is minimized by individually using Lingo 11.0 Solver, and the ideal solution and the nadir solution are obtained as $I = (0.5795, 38.0588, 72)$ and $N = (0.6817, 55.9759, 95)$, respectively. The decision makers determine the multi-choice goals as follows:

$$Z_1 = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} U(k) \geq 0.5795 \text{ and } \leq 0.6000,$$

$$Z_2 = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} C(k) \geq 38.0588 \text{ and } \leq 45,$$

$$Z_3 = C(n) \geq 72 \text{ and } \leq 90,$$

Subject to Equations 9-22.

The objective function values and multi-choice goals are normalized by dividing ($N_i - I_i$) values for $i = 1, 2, 3$. For example, the first objective function Z_1 is transformed to $Z_1 / (N_1 - I_1) = \sum_{k=1}^n w_j U(k) / 0.102$. The multi-choice goals are as follows:

$$f_1(x) = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} U(k) / 0.102 \geq 5.670 \text{ and } \leq 5.871,$$

$$f_2(x) = \sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} C(k) / 17.917 \geq 2.124 \text{ and } \leq 2.512,$$

$$f_3(x) = C(k) / 23 \geq 3.130 \text{ and } \leq 3.913,$$

Subject to Equations 9-22.

Therefore, the multi-choice goal programming is obtained by using the objective functions weights and multiple choice goals.

$$\min \sum_{i=1}^3 [(\beta + \alpha_i) d_i^+ + (\beta - \alpha_i) d_i^-],$$

Subject to

$$f_i(x) - d_i^+ + d_i^- = y_i, i = 1, 2, 3,$$

$$a_{i,min} \leq y_i \leq a_{i,max}, i = 1, 2, 3,$$

$$d_i^+, d_i^- \geq 0, i = 1, 2, 3,$$

Eqs. (9)-(22).

The value of parameter β is taken as 0.29 because this value should be less than the objective function weights and it should be a positive real number.

$$\min 0,69d_1^+ - 0,11d_1^- + 0,59d_2^+ - 0,01d_2^- + 0,59d_3^+ - 0,01d_3^-,$$

$$\sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} U(k) / 0.102 - d_1^+ + d_1^- = y_1,$$

$$5.670 \leq y_1 \leq 5.871,$$

$$\sum_{j=1}^n \sum_{k=1}^n w_j x_{jk} C(k) / 17.917 - d_2^+ + d_2^- = y_2,$$

$$2.124 \leq y_2 \leq 2.512,$$

$$C(k) / 23 - d_3^+ + d_3^- = y_3,$$

$$3.130 \leq y_3 \leq 3.913,$$

$$d_i^+, d_i^- \geq 0, i = 1, 2, 3,$$

y_1, y_2, y_3 : unrestricted variables

Eqs. (9)-(22).

The multi-choice goal programming model is solved by Lingo 11.0 Solver (Schrage, 2008) on an Intel (R) Core™ i7-2760QM CPU 2.40 GHz-based computer in a few seconds of computation time for the study. The values of negative and positive deviations are calculated as $d_1^+ = d_2^- = d_3^- = 0$; $d_1^- = 0.1896$; $d_2^+ = 0.1346$ and $d_3^+ = 0.0001$, respectively. The efficient objective function values are determined as 0.5795, 47.4196 and 90, respectively. The efficient sequence is obtained as 3-5-2-6-4-1. It means that the efficient point (0.5795, 47.4196, 90) obtained by using multi-choice goal programming falls into the multiple target values for the first and third multi-choice goals because $Z_1 = 0.5795 \in [0.5796, 0.6]$ and $Z_3 = 90 \in [72, 90]$. These values are satisfactory for the decision makers. On the other hand, the second multi-choice goal isn't satisfied by the obtained efficient point, because $Z_2 = 47.4196 \notin [38.0588, 45]$. The pessimistic selection of the multiple target values could be the cause of this result. If the decision makers

aren't satisfied with the obtained efficient point, they can change the multiple target intervals and/or the weights of the multi-choice goals. This allows the decision makers a flexible and attractive search over the Pareto surface.

Sensitivity analysis allows the effects of each change on the relative weights of the objective functions to be analyzed. A sensitivity analysis is performed for the different levels of the job weights for each job. As given in Table 9, the job weights belonging to the first row are obtained by the AHP method and the values of objective functions can be calculated as 0.5795, 47.4196 and 90, respectively. If the job weights have equal priority as the second row, the values of objective functions can be calculated as 0.5795, 47.4196 and 90, respectively. The different job weights are used to analyze the changes of the objective function values as shown in Table 9. It can be seen from Table 9 that the results are sensitive to the decision makers' preferences.

Table 9
The values of three objective functions according to the different job weights for each job

No	Job weights						Objective functions		
	w_1	w_2	w_3	w_4	w_5	w_6	Z_1	Z_2	Z_3
1	0.2182	0.2163	0.2200	0.1450	0.1022	0.0984	0.5795	47.4196	90
2	0.1667	0.1667	0.1667	0.1667	0.1667	0.1667	0.5795	47.4196	90
3	0.2107	0.0612	0.0163	0.1860	0.2547	0.2710	0.4579	49.3365	82
4	0.1612	0.1964	0.1073	0.2031	0.2069	0.1252	0.5607	46.5082	86
5	0.1114	0.2056	0.1440	0.3386	0.0998	0.1007	0.6556	44.7115	86
6	0.2331	0.0423	0.0893	0.0474	0.3626	0.2253	0.3228	45.3825	86
7	0.1075	0.1609	0.0224	0.1160	0.2314	0.3619	0.6158	37.0086	72
8	0.1971	0.2611	0.2304	0.1912	0.0694	0.0507	0.6494	54.0412	82
9	0.1251	0.2567	0.1524	0.3187	0.0257	0.1214	0.7005	49.3226	86
10	0.0512	0.2742	0.1697	0.0358	0.1241	0.3451	0.3612	44.6355	82

The efficient points are obtained by using the multi-choice goal programming according to the different levels of the weights of each objective function. Trade-offs among the objective function values based on the various weight combinations are given in Figure 3. If the weight of the weighted number of tardy jobs is greater than 0.30 and the other weights of the objectives are less than 0.60, then the first objective can achieve the ideal level as seen in Figure 3.

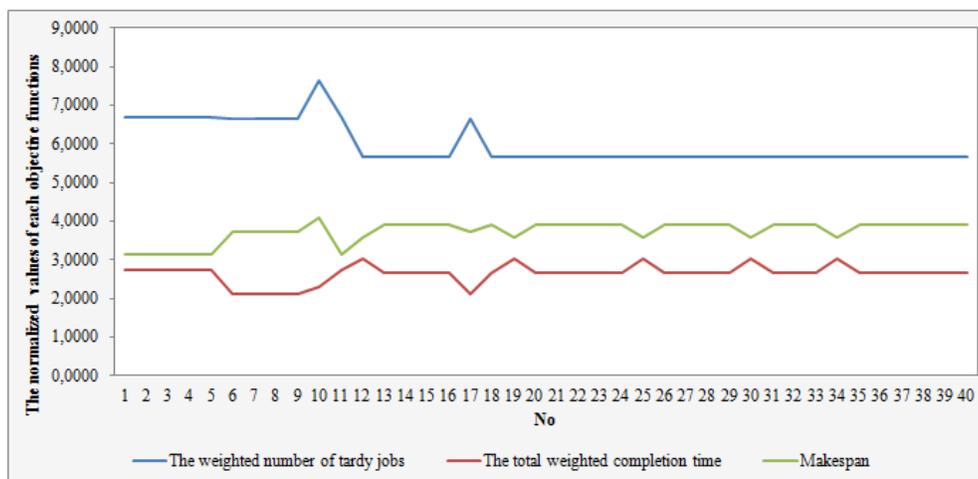


Figure 3. Efficient points related to the sensitivity analysis

3. Conclusion

In this paper, a methodology that consists of AHP, a 0-1 mixed integer non-linear programming model and multi-choice goal programming was proposed to schedule the jobs for a single machine with sequence-dependent setup times. The AHP allows a flexible multi-criteria decision making process by considering tangible and intangible criteria in a production scheduling environment. The scheduling constraints and the multiple objective functions for a single machine scheduling problem with sequence-dependent setup times were considered by using multi-objective programming. The multi-choice goal programming allows the decision maker to set multi-choice aspiration levels for each goal to obtain an efficient solution in the global region. Additionally, the multi-choice goal programming also guarantees an efficient solution for obtaining the assignment of candidates and reduces auxiliary constraints and additional variables.

The proposed approach was implemented with a small-size problem consisting of six jobs. A satisfactory solution was obtained by using the proposed approach. The sensitivity analysis was performed according to the job weights and the objective weights. The proposed methodology supports decision makers in the effective management of the job scheduling process. The proposed methodology can be applied to other fields such as personnel selection, logistics, machine selection, project management, portfolio management, etc. Job scheduling problems will become more complex in the near future because of the dynamic environment, and meta-heuristic methods can be required for large size problems.

REFERENCES

- Allahverdi A, Gupta J.N.D., & Aldowaisan T. (1999). A review of scheduling research involving setup considerations. *Omega*, 27, 219–39.
- Arcelus, F.J., & Chandra, R. (1983). On n-1-F setup dependent problems. *Engineering Optimization*, 7, 59–67.
- Armentano, V.A., & Mazzini, R. (2000). A genetic algorithm for scheduling on a single machine with set-up times and due dates. *Production Planning and Control*, 11, 713–720.
- Asano, M., & Ohta, H. (1999). Scheduling with shutdowns and sequence dependent set-up times. *International Journal of Production Research*, 37, 1661–1676.
- Bahalke, U., Yolmeh, A.M., & Shahanaghi, K. (2010). Meta-heuristics to solve single-machine scheduling problem with sequence-dependent setup time and deteriorating jobs, *International Journal of Advanced Manufacturing Technology*, 50, 749-759.
- Ben Mahmoud, H., Ketata, R., Ben Romdhane, T., & Ben Ahmed, S. (2010). Piloting a quality management system for study case using multi-choice goal programming, in: *Conference Proceedings – IEEE International Conference on Systems, Man and Cybernetics, 2010*, article no. 5641929, 2500–2505.
- Bianco, L., Mingozzi, A., & Ricciardelli, S. (1993). The travelling salesman problem with cumulative costs. *Networks*, 23, 81–91.
- Chang, T.Y., Chou, F.D. & Lee, C.E. (2004). A heuristic algorithm to minimize total weighted tardiness on a single machine with release dates and sequence-dependent setup times. *Journal of the Chinese Institute of Industrial Engineers*, 21, 289–300.
- Chang, C.T. (2007). Multi-choice goal programming. *Omega* 35, 389–396.
- Chang, C.T. (2008). Revised multi-choice goal programming. *Applied Mathematical Modelling*, 32, 2587–2595.
- Chen, C.L. (2012). Iterated hybrid metaheuristic algorithms for unrelated parallel machines problem with unequal ready times and sequence-dependent setup times. *International Journal of Advanced Manufacturing Technology*, 60, 693-705.
- Chen, J.F. (2013). Unrelated parallel-machine scheduling to minimize total weighted completion time. *Journal of Intelligent Manufacturing*, 1-14 (Article in Press).
- Da Silva A.F., Marins F.A.S., & Montevechi J.A.B. (2013a). Multi-choice mixed integer goal programming optimization for real problems in a sugar and ethanol milling company. *Applied Mathematical Modelling*, 37(9), 6146–6162.

- Da Silva A.F., Marins F.A.S., & Montevechi J.A.B. (2013b). Application of mixed binary goal programming in an enterprise in the sugar and energy sector. *Gestao e Producao*, 20(2), 321-335.
- Di, L., & Ze, T. (2011). A genetic algorithm with Tabu Search for multi-objective scheduling constrained flexible job shop, *Proceedings of 2011 Cross Strait Quad-Regional Radio Science and Wireless Technology Conference*, 1662-1665.
- Eren, T., & Güner, E. (2006). A bicriteria scheduling with sequence-dependent setup times. *Applied Mathematics and Computation*, 179, 378–385.
- Fang, Y., Wang, F., & Wang, H. (2011). Research of multi-objective optimization study for job shop scheduling problem based on grey ant colony algorithm. *Advanced Materials Research*, 308-310, 1033-1036.
- Fazlollahtabar, H., Mahdavi, I., Ashoori, M.T., Kaviani, S., & Mahdavi-Amiri, N. (2011). A multi-objective decision-making process of supplier selection and order allocation for multi-period scheduling in an electronic market. *International Journal of Advanced Manufacturing Technology*, 52, 1039–1052
- Fischetti, M., Laporte, G., & Martello, S. (1993). The delivery man problem and cumulative matroids. *Operations Research*, 41, 1055–1064.
- Franca, P.M., Mendes, A., & Moscato, P. (2001). A memetic algorithm for the total tardiness single machine scheduling problem. *European Journal of Operational Research*, 132, 224–242.
- Gagne, C., Price, W.L., & Gravel, M. (2002). Comparing an ACO algorithm with other heuristics for the single machine scheduling problem with sequence-dependent setup times. *Journal of the Operational Research Society*, 53, 895–906.
- Gupta, S.R., & Smith, J.S. (2006). Algorithms for single machine total tardiness scheduling with sequence dependent setups. *European Journal of Operational Research*, 175, 722–739.
- Kırıř, ř., & Saraç, T. (2009). Modeling the fuzzy goal programming for scheduling problem of single machine with sequence dependent set-up times. *1st International Fuzzy Systems Symposium, Ankara, Turkey*, 34-40.
- Lee, S.M., & Asllani, A.A. (2004). Job scheduling with dual criteria and sequence-dependent setups: mathematical versus genetic programming. *Omega*, 32 (2), 145–153.
- Lee, A.H.I., Kang, H.-Y., Yang, C.-Y., & Lin, C.-Y. (2010). An evaluation framework for product planning using FANP, QFD and multi-choice goal programming. *International Journal of Production Research*, 48(13), 3977–3997.
- Liao, C.N., & Kao, H.P. (2010). Supplier selection model using Taguchi loss function, analytical hierarchy process and multi-choice goal programming. *Computers and Industrial Engineering*, 58, 571–577.

- Lin, D., Lee, C.K.M., & Wu, Z. (2012). Integrating analytical hierarchy process to genetic algorithm for re-entrant flow shop scheduling problem. *International Journal of Production Research*, 50(7), 1813-1824.
- Macharis, C., Springael, J., De Brucher, K., & Verbeke, A. (2004). PROMETHEE and AHP: The design of operational synergies in multicriteria analysis. Strengthening PROMETHEE with ideas of AHP, *European Journal of Operational Research*, 153(2), 307–317.
- Mendes, A.S., Franca, P.M., & Moscato P. (2002). Fitness landscapes for the total tardiness single machine scheduling problem. *Neural Network World*, 12,165–180.
- Miyazaki, S., & Ohta H. (1987). Backward scheduling to minimize the actual mean flow time with dependent and independent setup times. *Proceedings of the. IXth ICPR*, 2680–2686.
- Paksoy, T., & Chang, C.T. (2010). Revised multi-choice goal programming for multi-period, multi-stage inventory controlled supply chain model with popup stores in Guerilla marketing. *Applied Mathematical Modelling*, 34, 3586–3598.
- Pinedo M. (1995). *Scheduling: theory, algorithms, and systems*. Englewood, NJ: Prentice-Hall, Inc.
- Panwalker S.S., & Iskander, W. (1977). A survey of scheduling rule. *Operations Research*, 25, 45–61.
- Rabadi, G., Mollaghasemi, M., & Anagnostopoulos G.C. (2004). A branch and- bound algorithm for the early/tardy machine scheduling problem with a common due-date and sequence dependent setup time. *Computers and Operations Research*, 31, 1727–1751.
- Romero, C. (2004). A general structure of achievement function for a goal programming model. *European Journal of Operational Research*, 153, 675–686.
- Saaty, T.L. (1980). *The Analytic Hierarchy Process*, New York: McGraw-Hill.
- Saaty, T.L. (2000). *Fundamentals of decision making and priority theory with the Analytic Hierarchy Process*. Pittsburgh, USA: RWS Publications.
- Schrage, L. (2008). *Optimization modeling with Lingo*, Chicago: Lindo Systems Inc.
- Shin, H.J., Kim, C.O., & Kim, S.S. (2002). A tabu search algorithm for single machine scheduling with release times, due dates, and sequence-dependent set-up times. *International Journal of Advanced Manufacturing Technology*, 19, 859–866.
- Sioud, A., Gravel, M. & Gagné, C. (2012). A hybrid genetic algorithm for the single machine scheduling problem with sequence-dependent setup times. *Computers and Operations Research*, 39, 2415-2424.

- Tan, K.C., & Narasimhan, R. (1997). Minimizing tardiness on a single processor with sequence-dependent setup times: A simulated annealing approach. *Omega*, 25, 619–634.
- Tan, K.C., Narasimhan, R., Rubin, P.A., & Ragatz, G.L. (2000). A comparison of four methods for minimizing total tardiness on a single processor with sequence dependent setup times. *Omega*, 28, 313–326.
- Ustun, O. (2012). Multi-choice goal programming formulation based on the conic scalarizing function. *Applied Mathematical Modelling*, 36, 974–988.
- Wang, L., & Wang, M. (1997). Hybrid algorithm for earliness–tardiness scheduling problem with sequence dependent setup time, *Proceedings of the IEEE Conference on Decision and Control* 2, 1219–1222.
- Witkowski, T., Antczak, P., & Antczak, A. (2009). Multi-objective decision making and search space for the evaluation of production process scheduling, *Bulletin of the Polish Academy of Sciences: Technical Sciences*, 57(3), 195-208.
- Wu, X., Sun, S., Yu, J., & Cai, Z. (2007). A multi -objective scheduling decision making model for the flexible job shop. *Zhongguo Jixie Gongcheng/China Mechanical Engineering*, 18(2), 161-165.