Contents lists available at GrowingScience

Management Science Letters

homepage: www.GrowingScience.com/msl

Minimizing total weighted tardiness for the single machine scheduling problem with dependent setup time and precedence constraints

Hamidreza Haddad[*] and **Mohammadreza Nematollahi**

Department of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran

ARTICLEINFO

Article history:
Received July 10, 2011
Received in Revised form
October, 12, 2011
Accepted 18 December 2011
Available online
24 December 2011

Keywords:
Scheduling
Single machine
Total weighted tardiness
Genetic algorithm
Precedence constraints

ABSTRACT

This paper tackles the single machine scheduling problem with dependent setup time and precedence constraints. The primary objective of this paper is minimization of total weighted tardiness. Since the complexity of the resulted problem is NP-hard we use metaheuristics method to solve the resulted model. The proposed model of this paper uses genetic algorithm to solve the problem in reasonable amount of time. Because of high sensitivity of GA to its initial values of parameters, a Taguchi approach is presented to calibrate its parameters. Computational experiments validate the effectiveness and capability of proposed method.

1. Introduction

This paper addresses the problem of sequencing *N* jobs on a single machine with sequence-dependent setup time and precedence constraint. The primary objective of this paper is to find a schedule of jobs, which minimizes the total weighted tardiness. Lawler (1997) showed that the problem of total weighted tardiness, even without setup time and precedence constraint, is NP-hard and cannot be solved in reasonable amount of time using traditional optimizing algorithms. Therefore, two groups of methods have been introduced to solve these kinds of problems including exact and near optimal solution methods. In the field of exact algorithms, these researches could be mentioned:

Xiaochuan and Feng (2006) proposed a branch and bound for minimizing total tardiness by considering the dependent setup time. Tan et al. (2000) solved the problem without considering the weight factor by using simulated annealing (SA), genetic algorithm (GA), Branch and Bound (B&B) and a new heuristic method. According to their results, B&B can reach the global optimum but it is

* Corresponding author. Tel: +9809126724147
E-mail addresses: hamidrhaddad@gmail.com (H. Haddad)

doi: 10.5267/j.msl.2011.12.020

not desirable when the number of jobs passes 25 due to its unreasonable computational time. On the other hand, Genetic search was found efficient for a great number of jobs.

On the other hand, there are many researchers who investigated heuristic and metaheuristic methods to generate near optimal solutions. Lee and Pinedo (1997) offered a constructive method named as Apparent Tardiness Cost with Setups (ATCS), which is able to solve problem instances with the small size in a reasonable run time. However, the efficiency of model for the large-scaled instances is not proper. Holsenback (1999) presented a heuristic, which could solve the problem for 50 jobs. Feo (1996) used greedy method and considered the distinct penalty for any lateness. Valente (2008) offered a new beam search where its efficiency and computational time were quite ideal for instances with low and medium size. Also, Sen (2003) discussed about the significance of weight factor of each job and showed that weight factor increases complexity of the problem exponentially. Furthermore, Cicirello (2003) accomplished a comprehensive research on tardiness problem and solved it using four different heuristics for 60 jobs. Allahverdi (1999) compared problems with dependent and independent setup time and concluded that although dependent setup time is more applicable for many industries, but it causes more additional computations in solving process.

Tasgetiren et al. (2009) offered a new algorithm named discrete differential evolution (DDE) that was similar to the greedy approach. Der Chou (2009) presented an experienced learning GA for the problem without setup time and compared their results with popular J.E Beasley’s OR library. In addition, Anghinolfi and paolucci (2009) offered a discrete PSO and a Memetic algorithm is used by Franka et al. (2001).

The hybrid algorithms have used in recent years to solve the scheduling problems to minimize total tardiness without setup time and weight factor. Generally, these classes of algorithms are organized into two groups containing Low-level and High-level hybrids. In Low-level hybrids, the first one, heuristic or metaheuristics, generates the initial solution and then the other algorithm tries to reform its performance whereas in high-level hybrids, one algorithm is known as the base and the other helps it for searching the neighborhood points without any changes on basic algorithm parameters. M’Hallah (2007) presented sets of high and Low-level hybrids based on dispatching rules, hill climbing, SA and a GA based evolutionary algorithm. Almeida and Centeno (1998) offered a compound algorithm using Tabu search, SA and Hill climbing to minimize total tardiness and Cheng et al. (2009) presented a hybrid based ant colony and four elimination rules which is classified as a high-level hybrid.

The remaining sections of this paper are organized as follows. In section 2, the problem is defined. In section 3, the solution methods are illustrated and finally the computational results are presented in section 4.

2. Problem formulation

In this paper, the single machine scheduling problem with *N* jobs with sequence dependent setup times is considered. The machine works with no idleness and all the jobs are ready for processing at time zero. We aim to find a sequence of jobs, which minimizes the total weighted of tardiness. The variables and parameters of problem are defined as follows:

N Number of jobs are ready to be scheduled
p_i The processing time of i^{th} job
w_i The weight factor of i^{th} job
T_i The tardiness of i^{th} job
c_i The completion time of i^{th} job
d_i The due date of i^{th} job

s_{ij} The setup time of job j when is processed exactly after job i

O_i The logical order of job i based on the precedence constraints

The basic model of the problem is as below:

$$\min \sum_{i=1}^{n} (W_i T_i) \tag{1}$$

subject to

$$\sum_j x_{ij} = 1 \tag{2}$$

$$c_0 = 0 \tag{4}$$

$$C_j \geq C_{j-1} + p_j + \sum_i \sum_{h \neq i} x_{ij} x_{h,j-1} S_{h,i} \tag{5}$$

$$T_i \geq \sum_{j=1}^{n} c_j x_{ij} - d_i \quad \forall \mathrm{i} \tag{6}$$

$$-1 + x_{i,j-1}.\, x_{kj} \leq \frac{O_i - O_k}{|O_i - O_k| + 1} \leq x_{i,j-1}.\, x_{kj} \tag{7}$$

$$T_i \geq 0 \quad \forall \mathrm{i} \tag{8}$$

where decision variables are as follows:

$$x_{ij} = \begin{cases} 1 & \text{If job } i \text{ is executed in priority } j \\ 0 & \text{Otherwise} \end{cases}$$

Eq. (1) introduces the objective function and tries to find a schedule of jobs to minimize the value of total weighted tardiness. Constraint (2) states that in each priority just one job could be planned. Constraint (3) assures that each job lies just in one priority. Constraint (4) mentions that machine is available from time zero. Constraint (5) declares that in any sequence, the completion time of proceed job is grater or equal to the sum of completion time of prior job, the processing times of scheduled proceed jobs and the setup times between them. Constraint (6) demonstrates the computation of tardiness. Vonstraint (7) describes the precedence constraint in which a job with higher order is not allowed to be processed before a job with lower order. Finally, constraint (8) states that the tardiness has a positive value.

3. Solution method

3.1. Genetic algorithm

Genetic algorithm firstly presented by Holland at University of Michigan and it is one of the most powerful Meta heuristics, which can be used as a useful tool for solving many optimization problems. The main idea of this approach is to use of natural genetic concepts and try to code the optimization problem using chromosomes and gens.

The fitness function is also described as below for each sequence:

$$f(i) = \frac{1}{\sum W_i T_i}, \tag{9}$$

where $w_i$demonstrates the weight factor of job i andT_i is the tardiness of the job located in i^{th} position in the sequence. We use of roulette wheel mechanism for selecting the suitable parents in each generation and two operators are used in order to obtain the neighborhood points of the current solution that are described follows:

- Mutation operator: in each generation, a random chromosome is selected using roulette wheel and then two random integer values are generated in [1-N] interval. Afterwards, the positions of generated numbers are changed in the sequence and the selected chromosome is replaced by this one in next generation.
- Cross over operator: in each generation, two chromosomes are selected using roulette wheel and the first one is considered as father and the other is as mother chromosome. Then a random number in [0-1] interval is selected. If the random number is less than 0.5 then the first job is selected of father chromosome and otherwise, the first job is selected from the mother and the selected job is located in a new chromosome (child chromosome). This process is iterated until all jobs are transformed into child chromosome. Afterwards, one of the parents is replaced by child randomly and the other is remained in the population for next generation.

In order to calibrate the proposed GA, a Taguchi approach is implicated. The input data for this approach is presented in Table 1.

Table 1
Important factors on the performance of GA

Factor	symbol	levels	Type	Degree of freedom
Number of chromosomes	A	3	A(1)=40 A(2)=50 A(3)=60	2
Mutation and cross over rates	B	3	B(1)=0.2 and 0.8 B(2)=0.5 and 0.8 B(3)=0.8 and 0.2	2
Selection mechanism	C	3	C(1)=roulette wheel C(2)= Best Selection C(3)=random selection	2
Number of iterations	D	3	D(1)=500 D(2)=1000 D(3)=1500	2

Based on the Taguchi matrix of orthogonal arrays, the L_9 pattern should be selected which fulfills the necessary requirement for designing the all experiments. The results for Taguchi experiment are depicted by Fig. 1.

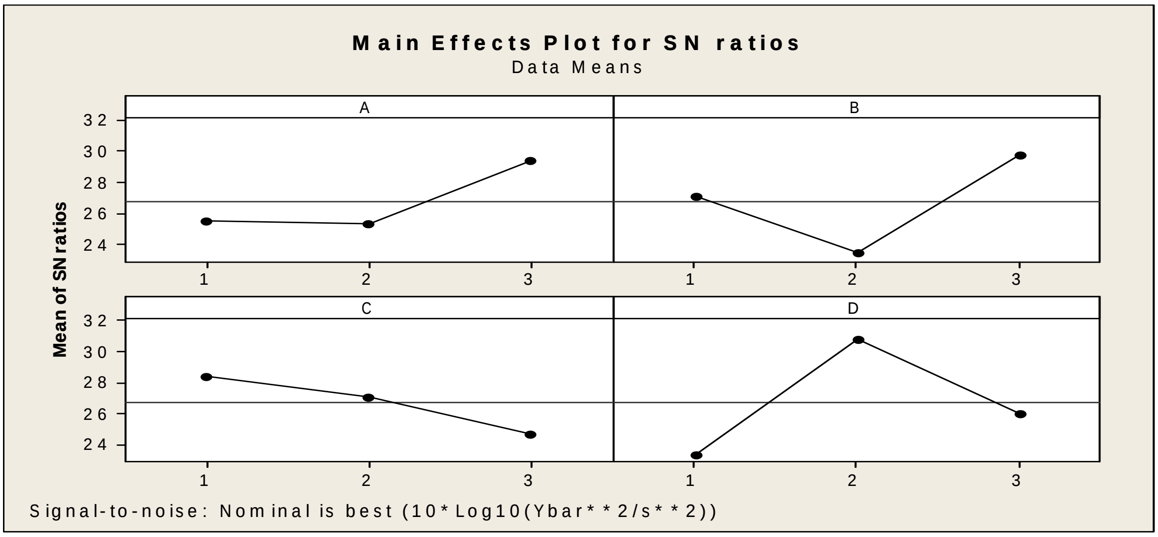

Fig. 1. Signal / Noise ratios of the results

The value of Signal/Noise ratio indicates how much a configuration of factor levels can minimize oscillation in algorithm performance while controlling other out of control factors. Therefore, a good configuration is the one with the greatest value of SNRA. This corresponds to the following configuration for the factors levels in GA:

A(3)=60, B(3)=0.8 and 0.2, C(1)=roulette wheel, D(2)=1000

On the other hand, Fig. 2 plots the mean of standard deviation of the observed best objective values. It is straightforward that a configuration with the minimum value of this index is the best. Hence, the following configuration for the factors levels is the most favorite in GA:

A(3)=60, B(3)=0.8 and 0.2, C(3)= Random selection, D(2)=1000

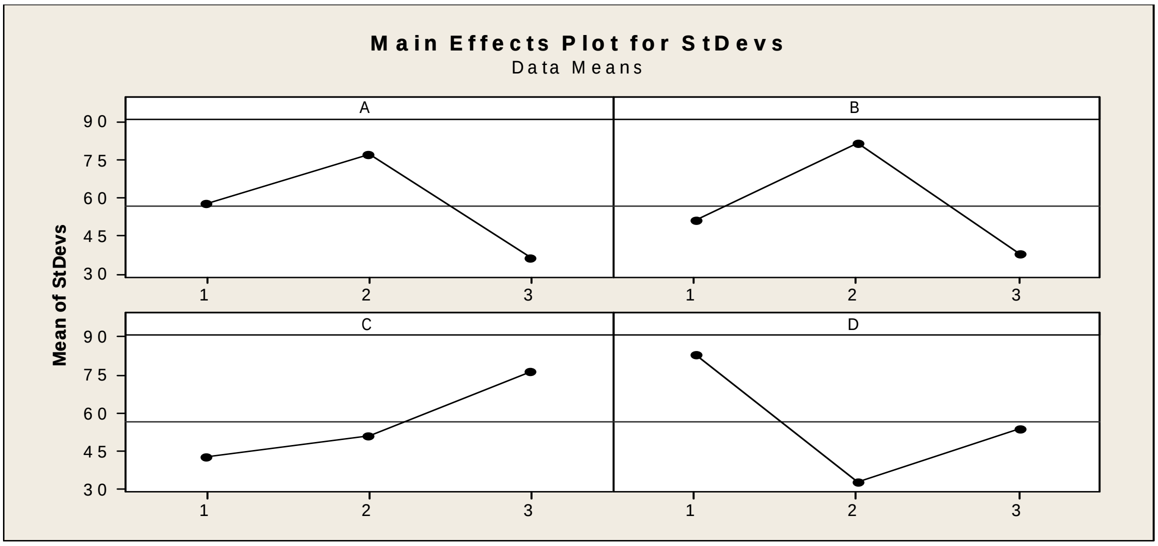

Fig. 2. Mean of standard deviation of the observed best objective values

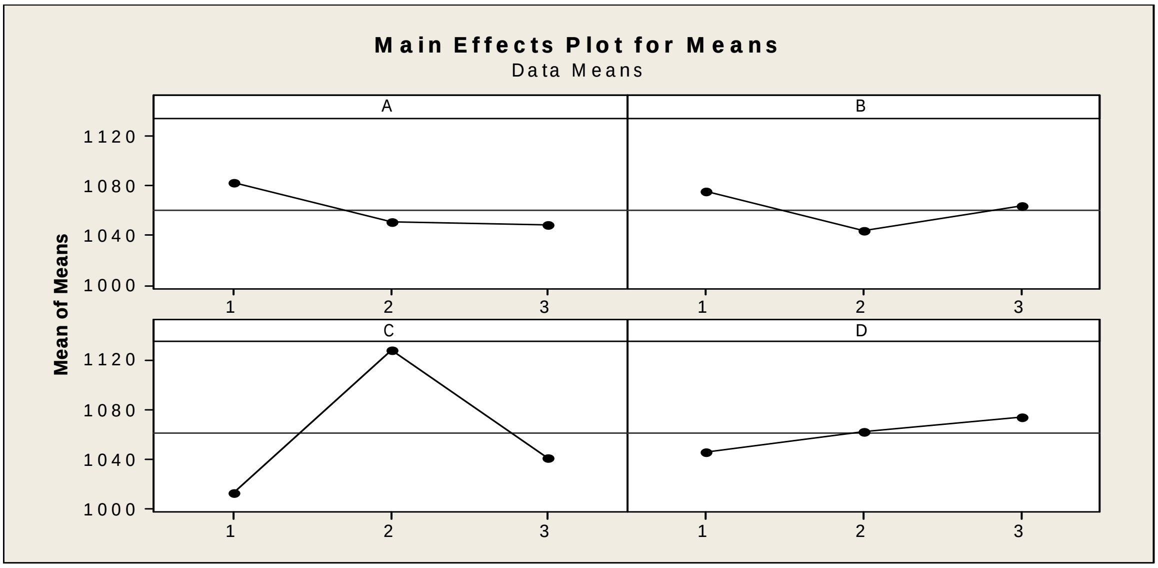

Fig. 3. Mean of the observed best objective values

Furthermore, Fig. 3 plots the mean of the observed best objective values. As, the problem objective is as minimization, a configuration of the factor levels corresponding to minimum value this index is the most favorite. This corresponds to the following configuration for factor levels of GA:
A(3)=60, B(2)=0.5 and 0.5, C(1)=roulette wheel, D(1)=500

Now, we should decide about the best configuration for factor levels of GA. This can be realized for each factor by counting number of the times, a specified level is introduced among the most favorite configurations. In this way, we can draw the following conclusion:

- A(3) = “60” is the best level of factor A, as it is the best reported level in three good configurations.
- B(3) = “0.8 and 0.2” is the best level of factor B, as it has the best reported robustness in terms of Signal / Noise ratio along with minimum standard deviation in three good configurations.
- C(1)= “roulette wheel” is the best level of factor C, as it has the best reported robustness in terms of Signal / Noise ratio along with minimum mean of the observed objective values in three good configurations.
- D(2)= “1000” is selected similar to C(1) for C.

4. Computational results

All the instances for the studied problem is coded by Visual Basic 6 on the system with CPU core i 7,1.6 GHZ and with 4 GB of RAM. the required values of parameters are generated according to Jolai et al. (2007) where processing times are selected from uniform distribution [1-100], weight factors from uniform distribution [1-10], setup times from uniform distribution [1-100] and finally due dates are generated from uniform distribution [0-ρ$\sum p_i$] and ρ is a controlling parameter for due dates and has the values of 0.2, 0.4 and 0.8.
To check the verification of proposed GA, the optimal solutions for some data sets of problem are obtained using Lingo 10 software and results are compared to the values of GA that are shown in table 5.

Table 5
The verification of proposed GA

N	Global optima		Proposed GA		
	VOF	Time (sec)	VOF	Time (sec)	%Error
4	433	4	433	4	0
6	933	26	933	9	0
8	1276	69	1276	14	0
10	2455	382	2455	16	0
12	-	-	3961	18	0
15	-	-	7633	19	0

Where first column shows the number of scheduled jobs, VOF demonstrates the value of objective function and %Error shows the percentage of error due to each approach that is calculated as:

$$\%Error = \frac{vof(GA) - vof(globaloptima)}{vof(globaloptima)} * 100 \tag{10}$$

This comparison shows that the proposed GA performs efficiently and presents the optimal solution till 10 jobs.

5. Conclusion and future researches

In this paper, the problem of single machine total weighted tardiness was considered and because of immense complexity, a genetic algorithm was presented so as to obtain near optimal solution in reasonable time. In order to calibrate the parameters of proposed GA, a Taguchi approach was accomplished. In computational study section, the problem was solved using various data sets that presented GA could present the optimal solutions till 10 jobs. For the future research, the problem could be considered by adding more constraints and assumptions including preemption, release date and etc.

Acknowledgment
The authors would like to thank the anonymous referees for the valuable comments on earlier version of this work.

References

Allahverdi, A., Gupta, J.N.D., & Aldowaisan, T.A. (1999). A review of scheduling research involving setup considerations. *Omega*, 27, 219-239.

Almeida, MT., & Centeno, M. (1998). A composite heuristic for the single machine early–tardy job scheduling problem. *Computers & Operations Research*, 25, 625–35.

Anghinolfi, D., & Paolucci, M. (2009). A new discrete particle swarm optimization approach for the single-machine total weighted tardiness scheduling problem with sequence-dependent setup times. *European Journal of Operational Research*, 193, 73–85

Aarts, E., & Lenstra, J.K. (1997). *Local Search in Combinatorial Optimization*. New York:John Wiley & Sons.

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.

Cheng, T.C.E., Lazarev, A., & Gafarov, E.R. (2009). A hybrid algorithm for the single-machine total tardiness problem. *Computers & Operations Research*, 36, 308 – 315.

Cicirello, V.A. (2003). Weighted tardiness scheduling with sequence-dependent setups. New York: A benchmark library. Technical Report, Intelligent Coordination and Logistics Laboratory, Robotics Institute, Carnegie Mellon University.

Der Chou, F. (2009). An experienced learning genetic algorithm to solve the single machine total weighted tardiness scheduling problem. *Expert Systems with Applications*, 36, 3857–3865.

Feo, T.A., Sarathy, K., & McGahan, J. (1996). A GRASP for single machine scheduling with sequence dependent setup costs and linear delay penalties. *Computers & Operations Research*, 23, 881–895.

Franca, P.M., Mendes, A., & Moscato, P. (2001). A mimetic algorithm for the total tardiness single machine scheduling problem. *European Journal of Operational Research*, 132, 224–42.

Holsenback, J.E., Russell, R.M., Markland, RE., & Philipoom, P.R. (1999). An improved heuristic for the single-machine.weighted-tardiness problem, *Omega*, 27, 485–495.

Jolai, F., Rabbani, M., Amalnick, S., Dabaghi, A., Dehghan, M., & YazadnParast, M. (2007). Genetic algorithm for bi-criteria single machine scheduling problem of minimizing maximum earliness and number of tardy jobs. *Applied Mathematics and Computation*, 194, 552–560

Khorshidian, H., Javadian, N., Zandieh, M., Rezaeian, J., & Rahmani, K. (2011). A genetic algorithm for JIT single machine scheduling with preemption and machine idle time. *Expert Systems with Applications*, 38, 7911–7918.

Lawler, E.L. (1997). A pseudo polynomial algorithm for sequencing jobs to minimize total tardiness. *Annals of Discrete Mathematics*, 1, 331–342.

Lee, Y.H., Bhaskaram, K., & Pinedo, M. (1997). A heuristic to minimize the total weighted tardiness with sequence-dependent setups. *IIE Transactions*, 29, 45–52.

Xiaochuan, L., & Feng, Ch. (2006). A branch and bound algorithm of the single machine schedule with sequence dependent setup times for minimizing total tardiness. *Applied Mathematics and Computation*, 183, 575–588.

M’Hallah, R. (2007). Minimizing total earliness and tardiness on a single machine using a hybrid heuristic. Computers & Operations Research, 34, 3126 – 3142.

Sen, T., Sulek, J.M., & Dileepan, P. (2003). Static scheduling research to minimize weighted and unweighted tardiness: a state-of-the-art survey. *International Journal of Production Economics*, 83, 1–12.

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-International Journal of Management Science*, 28, 313–326.

Tasgetirena, MF., Quan-Ke, P., & Yun-Chia, L. (2009). A discrete differential evolution algorithm for the single machine total weighted tardiness problem with sequence dependent setup times. *Computers & Operations Research*, 36, 1900-1915.

Valente, J.M.S., & Alves, R. (2008). Beam search algorithms for the single machine total weighted tardiness scheduling problem with sequence-dependent setups. *Computers & Operations Research*, 35, 2388–2405.

Van Laarhoven, P.J.M., & Aarts, E.H. (1988). *Simulated Annealing: Theory and Applications*. Dordrecht: Kluwer Academic Publishers.