SINGLE MACHINE SCHEDULING PROBLEM WITH PRECEDENCE CONSTRAINTS AND DETERIORATING JOBS

N. Hamta, S.M.T. Fatemi Ghomi*

Department of Industrial Engineering, Amirkabir University of Technology
P.O. Box 15875-4413, Tehran, Iran
nima.hamta@aut.ac.ir - fatemi@aut.ac.ir

U. Bahalke and H.R. Golpaigani

Department of Industrial Engineering, Iran University of Science and Technology
P.O. Box 16765-163, Tehran, Iran
unesbahalke@ind.iust.ac.ir - hrgolpaigani@ind.iust.ac.ir

*Corresponding Author

(Received: November 08, 2009 – Accepted in Revised Form: June 21, 2011)

Abstract This paper considers the single machine scheduling problem with precedence constraints and deteriorating jobs and a mathematical model based on binary integer programming (BIP) is developed. Since the precedence constraints exist, a job cannot start before completion of its all predecessors. The proposed model consists of two steps: in the first step, the earliest starting time of each job is computed; then the results are used in the second step in which an optimal sequence between jobs is determined with the aim of minimizing the total completion time. Finally, a numerical example is presented and solved using optimization software LINGO 8.0.

Keywords Single Machine, Scheduling, Precedence Constraints, Job Deterioration, Mathematical Modeling

چکیده این مقاله به مسئله زمانبندی تک ماشینه با محدودیت‌های پیش نیازی و کارهای زوال‌پذیر می‌پردازد. برای این منظور، یک مدل ریاضی براساس برنامه‌ریزی عدد صحیح باینری توسعه داده شده است. وجود محدودیت‌های پیش نیازی در مسئله سبب می‌شود که کارها نتوانند پیش از تکمیل تمامی پیش‌نیازهای خود آغاز شوند. مدل در دو مرحله ارائه شده است: در مرحله اول زودترین زمان شروع هر کار محاسبه می‌شود. سپس در دومین گام، یک توالی بهینه بین کارها با هدف حداقل کردن زمان کل تعیین می‌شود. در انتها یک مثال عددی ارائه و با نرم‌افزار LINGO 8.0 حل شده است.

1. INTRODUCTION AND LITERATURE REVIEW

The single machine scheduling problem has been widely enriched over the past years. The aim of most researches is to approach the models to the real world situations. In this paper, the deteriorating jobs are concerned to schedule the jobs with the existence of precedence constraints. In many realistic applications, jobs may deteriorate, while waiting to be processed. For example, a drop in the temperature of an ingot, while waiting to enter the rolling machine, requires the ingot to be reheated before rolling.

1.1. Review on deteriorating job scheduling problems The deteriorating jobs were first introduced by Browne and Yachiali [1]. They defined the processing times for this type of jobs as a linear function of their starting time $(a_j + b_j.St_j)$, where a_j is the fixed part of processing time, b_j the deterioration growth rate, and St_j the starting time of job j. Since then, deteriorating job scheduling problems have been widely discussed. Oron [2] studied a single machine scheduling problem with simple linear deterioration. He assumed that job processing time is a simple linear function of a job-dependent growth rate and the job starting time and tried to minimize the total absolute deviation of completion times (TADC).

Low et al. [3] investigated the non-preemptive case with the availability constraint and the objective function of minimization of the make-spam. They developed a 0-1 integer programming model and showed that the problem is NP-hard. Then, they proposed some heuristics based on the bin packing concepts and a good maintenance starting strategy when half the jobs have been already processed. Raut et al. [4] introduced a capacitated single machine case and proposed several heuristics based on a multiplicative piece-wise metric as an approximation of the slope of job value deterioration. Wang [5] considered the single machine scheduling problem with the effects of learning and deterioration, i.e. the processing time of jobs are defined by functions of their starting times and positions in the sequence. He proved that by considering the learning effect and deteriorating jobs, single-machine make-spam and sum of completion times (square) minimization problems remain polynomial solvable. Furthermore, he showed that the WSPT and EDD rules can construct the optimal sequence for the weighted sum of completion times and the maximum lateness, respectively.

Cheng and Sun [6] studied several single machine scheduling problems in which the processing time of each job was a linear function of its starting time (deterioration) and jobs could be rejected by paying penalties. The objectives were minimizing the make-spam, the total weighted completion time and the maximum lateness/tardiness plus the total penalty of the rejected jobs. They showed that all these problems are NP-hard and also proposed algorithms based on dynamic programming including pseudo-polynomial time optimal algorithms and fully polynomial time approximation schemes to solve the problem. Lee et al. [7] investigated the problem of *m*-machine permutation flow shop which aims to minimize the total completion time and studied several deterioration patterns. They also proposed a dominance rule and a lower bound to make an efficient branch-and-bound algorithm. Ji and Cheng [8] studied a parallel machine scheduling problem in which the processing time of each job was a simple linear function of its starting time. They developed a fully polynomial-time approximation scheme for the case with *m*-machines. Lee et al. [9] considered a two-machine flowshop scheduling problem to minimize the make-spam in which job processing times defined as increasing functions of their starting times. They proposed an exact algorithm to solve the problems in a reasonable time and three heuristics to obtain the near optimal solutions. They conducted a simulation study to evaluate the performance of the proposed heuristics and investigated the impact of the different deterioration rates.

1.2. Review on precedence constrained scheduling problems Recently, some researchers have taken into account the scheduling problems with precedence constraints. In many realistic situations due to technological and/or organizational requirements, jobs cannot be carried out in an arbitrary sequence and a set of precedence relations specify the permissible orders of performing the jobs. Few studies have considered precedence constrained scheduling problems. Kim et al. [10] proposed a heuristic procedure for parallel machine scheduling problems where multiple jobs with s-precedence constraints are processed on multiple identical parallel machines. Hanen and Kordon [11] compute periodic cyclic schedules for linear precedence constraints graphs with a linear programming model. They presented a polynomial algorithm with the aim of minimizing the maximal period of a task.

Since some real world problems are usually simplified in the solution process with temporal constraints, Bredström and Rönnqvist [12] used combined vehicle routing and scheduling model with temporal precedence and synchronization constraints. They compared a direct use of a commercial solver with a proposed heuristic method. Giroudeau et al. [13] presented a scheduling approximation to schedule unitary tasks with precedence constraint and large communication delays. They used a polynomial-time approximation algorithm.

Lo et al. [14] proposed a modified ant colony optimization approach to solve the scheduling problems with precedence and resource constraints. Cakar et al. [15] used genetic algorithm (GA) and simulated annealing (SA) method for minimizing the mean tardiness, when jobs with ***n***-number of precedence constraints are assigned on ***m***-number of parallel robots. They showed that GA gives better results than SA. Pedersen et al. [16] developed a solution method for minimizing make-spam of a practical large-scale scheduling problem with elastic jobs. They presented a new method to approximate the server exploitation of the elastic jobs, while the jobs are processed on three servers and restricted by precedence constraints, time windows and capacity limitations. They solved the problem using a tabu search algorithm.

Lushchakova [17] suggested an O(n^2) algorithm for a scheduling problem with two identical parallel machines, equal processing times and precedence constraints. Coll et al. [18] introduced a new integer programming formulation for the problem of multiprocessor scheduling under precedence constraints. Ramachandra and Elmaghraby [19] developed a binary integer program (BIP) and a dynamic program (DP) for sequencing precedence related jobs on two machines problem and then proposed a genetic algorithm (GA) for solving different problem sizes.

In this paper, single machine scheduling problem with deteriorating jobs and precedence constraints between jobs are studied. Due to the existence of precedence constraints, some jobs are not available at time zero. In other words, a job is available when all of its predecessors have been processed. On the other hand, jobs deteriorate while waiting to be processed. The earliest available time of job j is defined as the minimum completion time of all predecessors, if the predecessors are processed before it. Thus, the earliest available time of each job should be computed firstly, and then the results are used to determine a sequence for the jobs with the aim of minimizing the make-spam.

The paper is organized as follows. In Section 2, the problem is defined, the assumptions and notations are introduced, and the mathematical formulation of the model is developed. In Section 3, a numerical example is provided to elaborate the performance of the proposed model. Finally, Section 4 is devoted to conclusions and recommendations for future works.

2. PROBLEM DESCRIPTION

Let there be N jobs, $J_1, J_2, ..., J_N$ in the presence of precedence relations between jobs. The earliest starting time (available time) of each job is dependent on its predecessors. The processing time of each job is defined as $p_j = a_j + b_j.(St_j - Es_j)$, where a_j is the fixed part of the processing time, b_j is the growth rate of the deterioration, St_j is the starting time of job j, and Es_j is the earliest starting time of job j. According to the above-mentioned relation, when there is a delay in starting time of a job, the processing time of that job increases. Due to the existence of precedence constraints (s-precedence), a successor cannot be processed before all its predecessors are performed completely.

2.1. Assumptions The addressed problem in this paper is characterized as follows:

1. As mentioned before, the processing time of each job is defined as: $p_j = a_j + b_j.(St_j - Es_j)$.
2. The jobs are dependent on each other.
3. The deterioration growth rate is independent of the machines.
4. The earliest starting time of job j (Es_j) is the minimum total completion time of all its predecessors, if only these jobs are processed before job j. In fact, the earliest start time for each job is calculated separately and it assumes that just all the predecessors of that job are done before.
5. No job preemption is allowed.
6. Two dummy jobs are defined, one of them is assigned to the first sequence, and the other is assigned to the last sequence. The processing time of these jobs is zero.

2.2. Notations Before the mathematical formulation is presented, we introduce the following notations:

N	the number of all jobs
n_i	the number of predecessors of job i
Ep_i	set of immediate predecessors of job i
Jp	set of jobs that have predecessor
$Apr_{(i)}$	set of all predecessors of job i
i, j	$j \in \{0,1,...,N+1\}$ designate the job. Job 0 and job N+1 are dummy jobs, which are always at the first and at the last sequence positions, respectively.
St_j	starting time of job j, $j \in \{0,1,...,N+1\}$
p_j	processing time of job j, $j \in \{0,1,...,N+1\}$
C_j	completion time of job j
b_j	the growth rate of the processing time of job j
a_j	fixed part of the processing time of job j
Es_j	the earliest starting time of job j
M	a large positive number

Decision variable

$$x_{ij} = \begin{cases} 1 & \text{if job } j \text{ immediately follows job } i \\ 0 & \text{otherwise.} \end{cases}$$

2.3. Mathematical formulation According to the aforementioned notations, we have the following formulation:

Objective functions

$$\text{Min } Z_1 = \sum_{j=0}^{N+1}\sum_{i=0}^{N+1}(x_{ij} \times p_j) \tag{1}$$

$$\text{Min } Z_2 = \sum_{g=0}^{n_g+1}\sum_{r=0}^{n_r+1}(x_{rg} \times p_g) \tag{2}$$

$$r \in Apr_{(r)},\ g \in Apr_{(g)}$$

Subject to

$$\sum_{i=0}^{N+1} x_{ij} = 1 \tag{3}$$

$$j = 1,2,...,N+1, i \neq j$$

$$\sum_{j=1}^{N} x_{i-1,j} = 1 \tag{4}$$

$$i = 1, 2, \ldots, N+1,\ i \neq j+1$$

$$\sum_{i=1}^{N} x_{i0} = 0 \tag{5}$$

$$St_j + M(1 - x_{ij}) \geq C_i \tag{6}$$

$$i, j = 1,2,...,N+1,\ i \neq j$$

$$p_j = a_j + b_j \times (St_j - Es_j) \tag{7}$$

$$j = 1, 2, \ldots, N+1$$

$$C_j \geq St_j + p_j \tag{8}$$

$$j = 1, 2, \ldots, N+1$$

$$St_i \geq p_r + St_r \tag{9}$$

$$r \in Ep_i, i \in Jp$$

$$St_0 = 0,\ Es_{N+1} = 0,\ Es_0 = 0$$

$$C_0 = 0$$

$$C_i,\ St_j,\ P_j \geq 0$$

$$x_{ij} \in \{0,1\}$$

In the above model, Relation (1) is the objective function that aims to minimize make-spam. Relation (2) is used to compute the earliest starting time of each job. Constraint (3) guarantees that always one job, job i, is assigned before job j. Constraint (4) states that each job (except the last job), immediately is followed by one job, job j. Constraint (5) ensures that a dummy job (i=0) is assigned to the first sequence position. Constraint (6) states that if job i is followed by job j, the starting time of job j will be greater than or equal to the completion time of job i. Relation (7) calculates the processing time of each job via a linear function of starting time, fixed part and the growth rate of its processing time. Constraint (8) expresses that the completion time of each job is at least equal to the sum of its starting time and processing time. Constraint (9) ensures that the staring time of each job is greater than or equal to the completion time of its predecessors.

Note that, by considering n_i as the number of predecessors of job *i*, the earliest starting time of that job is computed in the first step (Step 1). The obtained results of Step 1 are used as input data to compute the earliest starting time of all jobs. Then, the best sequence of jobs is determined to minimize the make-spam (Step 2).

2.4. Solution procedure In this subsection, the solution procedure of the problem is described with a flowchart given in Figure 1. In the first step of the single objective model, the earliest starting time of all jobs are calculated separately. This step is divided into several stages on the basis of the number of existing jobs. In each stage while calculating the earliest starting time of job *i*, a detailed sub-problem from the basic original problem is constructed in which the existing jobs are direct or indirect predecessors of job *i* with the objective function of make-spam. The optimum make-spam of the sub-problem is obtained when job *i* is performed at the earliest starting time in the sequence. Then in step 2, the calculated earliest starting time of all jobs is assigned to Relation (7) and the model proceeds to obtain the optimum value of make-spam for the basic original problem. Next section gives a numerical example to clearly demonstrate the solution process.

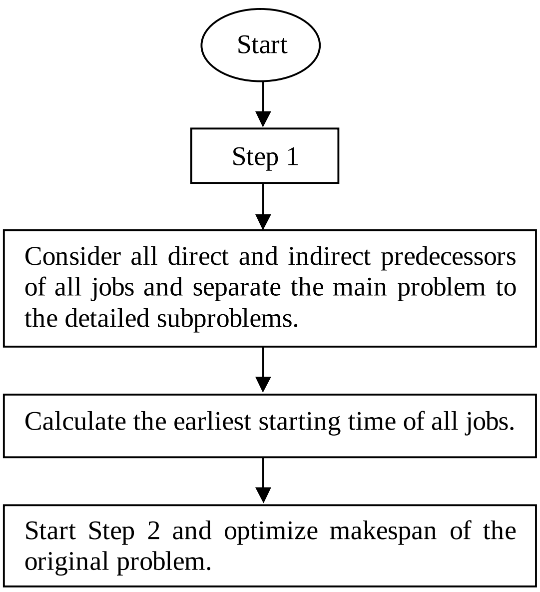

Figure 1. Flowchart of the solution procedure.

3. NUMERICAL EXAMPLE

This section presents a numerical example with 8 jobs to clarify the developed model and the solution process. The corresponding precedence graph is given in Figure 1. The aim is to assign these jobs in a sequence so that the make-spam of the problem is minimized. The earliest starting time of jobs with no predecessor is 0. For example as it is shown in Figure 2, job 1 has no predecessor; therefore, its earliest starting time is 0. Tables 1 and 2 report the input data of the problem. Table 3 presents the results of the first step. For example in stage 5, only the predecessors of job 6 are considered and a sequence with the aim of minimizing total completion time is determined by considering the earliest starting time of jobs.

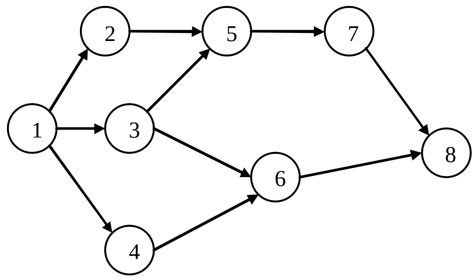

Figure 2. Precedence diagram for a numerical example with 8 jobs.

TABLE 1. The Fixed Part of the Processing Time (a_j).

Job No.	1	2	3	4	5	6	7	8
a_j	2	3	4	3	3	2	3	3

TABLE 2. The Growth Rate of the Processing Time (b_j).

Job No.	1	2	3	4	5	6	7	8
b_j	0.25	0.3	0.45	0.5	0.25	0.4	0.65	0.7

TABLE 3. The Results of Step 1.

Stage	The sequence of all predecessors of job i	Value of Es	Job i
1	$d_0 \rightarrow 1 \rightarrow d_2$	2	2
2	$d_0 \rightarrow 1 \rightarrow d_2$	2	3
3	$d_0 \rightarrow 1 \rightarrow d_2$	2	4
4	$d_0 \rightarrow 1 \rightarrow 2 \rightarrow d_3$	5	5
5	$d_0 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow d_4$	10.35	6
6	$d_0 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow d_5$	14.5	7
7	$d_0 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow$ $6 \rightarrow 2 \rightarrow 5 \rightarrow 7 \rightarrow d_8$	34.52594	8

In the solution procedure, two dummy jobs are defined: one is for the first sequence and the other is for the last sequence so that the model would be defined correctly. In order to demonstrate the solution procedure clearly, the precedence diagrams of stages 5, 6 and 7 are illustrated in Figures 3-5, respectively. Tables 4 and 5 show the final results of the model. In this paper, all models have been solved using optimization software LINGO 8.0.

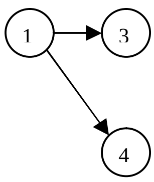

Figure 3. Precedence diagram of stage 5.

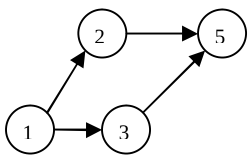

Figure 4. Precedence diagram of stage 6.

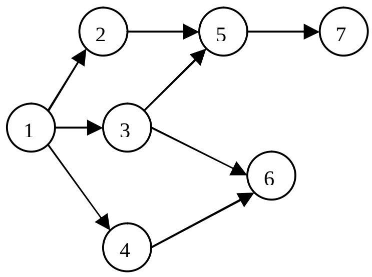

Figure 5. Precedence diagram of stage 7.

TABLE 4. The Final Results of the Model.

Job	Starting time of job (j)	Processing time of job (j)
1	0	2
2	12.35	6.105
3	5	5.35
4	2	3
5	18.455	6.36375
6	10.35	2
7	24.81875	9.707187
8	34.52594	3

TABLE 5. Optimum Sequence of Jobs and Corresponding Make-spam.

The optimum sequence of jobs
$d_0 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 6 \rightarrow 2 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow d_9$
The value of objective function (make-spam)
37.52594

4. CONCLUSIONS AND FUTURE WORKS

Single machine scheduling is the process of assigning a group of jobs to a single machine or resource. The jobs are arranged so that one or more performance measures may be optimized. This problem has been widely enriched over the past few years with many realistic approaches and much effort has been made to reduce the distance between the academic theory and the industrial reality. Relating to this issue, this paper addressed the single machine scheduling problem with precedence constraints and deteriorating jobs. A mathematical model based on 0-1 integer programming was developed. The proposed model consists of two steps: in the first step, the earliest starting time of each job is computed, and then the results are used in the second step to determine an optimal sequence for jobs so that the total completion time of the model (make-spam) is minimized. To demonstrate the solution procedure of the model, a numerical example was explained in details.

Development of another exact solution technique (e.g., branch-and-bound method) or an effective metaheuristic algorithm (e.g., genetic algorithm or simulated annealing) to solve relatively large scale problems is a challenging area for future works.

5. REFERENCES

1. Browne, S. and Yechiali, U., "Scheduling Deteriorating Jobs on a Single Processor", ***Operations Research***, Vol. 38, (1990), 495-498.
2. Oron, D., "Single Machine Scheduling with Simple Linear Deterioration to Minimize Total Absolute Deviation of Completion Times", ***Computers & Operations Research***, Vol. 35, (2008), 2071-2078.
3. Low, C., Hsua, C.J. and Su C.T., "Minimizing the Make-spam with an Availability Constraint on a Single Machine under Simple Linear Deterioration", ***Computers & Mathematics with Applications***, Vol. 56, (2008) 257-265.
4. Raut, S., Swami, S. and Jatinder, N.D., "Scheduling a Capacitated Single Machine with Time Deteriorating Job Values", ***International Journal of Production Economics***, Vol. 114, (2008), 769-780.
5. Wang, J.B., "Single-Machine Scheduling Problems with the Effects of Learning and Deterioration", ***Omega***, Vol. 35, (2007), 397-402.
6. Cheng, Y. and Sun, S., "Scheduling Linear Deteriorating Jobs with Rejection on a Single Machine", ***European Journal of Operational Research***, Vol. 194, (2009), 18-27.
7. Lee, W.C., Wu, C.C., Chung, Y.H. and Liu, H.C., "Minimizing the Total Completion Time in Permutation Flow Shop with Machine-Dependent Job Deterioration Rates", ***Computers & Operations Research***, Vol. 36, (2009), 2111-2121.
8. Ji, M. and Cheng, T.C.E., "Parallel-Machine Scheduling with Simple Linear Deterioration to Minimize Total Completion Time", ***European Journal of Operational Research***, Vol. 188, (2008), 342-347.
9. Lee, W.C., Wu, C.C., Wen, C.C. and Chung, Y.H., "A Two-Machine Flowshop Make-spam Scheduling Problem with Deteriorating Jobs", ***Computers & Industrial Engineering***, Vol. 54, (2008), 737-749.
10. Kim, E.S., Sung, C.S. and Lee, I.S., "Scheduling of Parallel Machines to Minimize Total Completion Time Subject to s-Precedence Constraints", ***Computers & Operations Research***, Vol. 36, (2009), 698-710.
11. Hanena, C. and Kordon, A.M., "Periodic Schedules for Linear Precedence Constraints", ***Discrete Applied Mathematics***, Vol. 157, (2009), 280-291.
12. Bredström, D. and Rönnqvist, M., "Combined Vehicle Routing and Scheduling with Temporal Precedence and Synchronization Constraints", ***European Journal of Operational Research***, Vol. 191, (2008), 19-31.
13. Giroudeau, R., Konig, J.C., Moulai, F.K. and Palaysi, J., "Complexity and Approximation for Precedence Constrained Scheduling Problems with Large Communication Delays", ***Theoretical Computer Science***, Vol. 401, (2008), 107-119.
14. Lo, S.T., Chen, R.M., Huang, Y.M. and Wu, C.L., "Multiprocessor System Scheduling with Precedence And Resource Constraints Using an Enhanced Ant Colony System", ***Expert Systems with Applications***, Vol. 34, (2008), 2071-2081.
15. Cakar, T., Koker, R. and Demir H.I., "Parallel Robot Scheduling to Minimize Mean Tardiness with Precedence Constraints Using a Genetic Algorithm", ***Advances in Engineering Software***, Vol. 39, (2008), 47-54.
16. Pedersen, C.R., Rasmussen R.V. and Andersen K.A., "Solving a Large-Scale Precedence Constrained Scheduling Problem with Elastic Jobs Using Tabu Search", ***Computers & Operations Research***, Vol. 34, (2007), 2025-2042.
17. Lushchakova, I.N., "Two Machine Preemptive Scheduling Problem with Release Dates, Equal Processing Times and Precedence Constraints", ***European Journal of Operational Research***, Vol. 171, (2006), 107-122.
18. Coll, P.E., Ribeiro, C.C. and Souza, C.C.D., "Multiprocessor Scheduling under Precedence Constraints: Polyhedral Results", ***Discrete Applied Mathematics***, Vol. 154, (2006), 770-801.
19. Ramachandra, G. and Elmaghraby, S.E., "Sequencing Precedence-Related Jobs on Two Machines to Minimize the Weighted Completion Time", ***International Journal of Production Economics***, Vol. 100, (2006), 44-58.