ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(31), DOI: 10.17485/ijst/2016/v9i31/96259, August 2016* *ISSN (Online) : 0974-5645*

Single Machine Scheduling Model for Total Weighted Tardiness

Neelam Tyagi[1], R. P. Tripathi[1] and A. B. Chandramouli[2]

[1]Department of Mathematics, Graphic Era University, Dehradun – 248002, Uttarakhand, India; neelam24tyagi@gmail.com, tripathi_rp0231@rediffmail.com
[2]Department of Mathematics, Meerut College, Meerut – 250001, Uttar Pradesh, India; dr.abchandramouli@gmail.com

Abstract

Objectives: We proposed two heuristic algorithms for Total Weighted Due Date Tardiness Scheduling (TWDDTS). The main aim of this paper is to optimize the weighted tardiness based criteria. **Methods/Statistical Analysis:** The first one heuristic algorithm for 'TWDDTS' is based on dispatching rule (EDD-Earliest Due Date) and the second one heuristic algorithm for Modified Total Weighted Due Date Tardiness Scheduling (MTWDDTS) is based on modified weighed due dates. We equated the effectiveness of both the proposed heuristic algorithms with the help of numerical illustrations. **Findings:** The main aim to propose these heuristic algorithms is to obtain the optimal solution of the problem related to weighted tardiness scheduling based criteria, when processing times of the jobs are also associated with probabilities. These heuristic algorithms are justified by numerical illustrations and comparative study of both the heuristic algorithms (with the help of numerical example) show that an improved heuristic algorithm for MTWDDTS is outperform and give better results than a TWDDTS heuristic algorithm. We also found that when we measure the mean completion time of jobs by both the heuristic algorithms then we observed that the MTWDDTS heuristic algorithm gives the best result as compared to a TWDDTS heuristic algorithm. So we find that MTWDDTS gives the best result for minimization the weighted tardiness based criteria as well as makespan. **Application/Improvements:** The proposed MTWDDTS algorithm is more useful than the EDD dispatching rule for weighted tardiness based scheduling problems. It is easy to understand and provide an important tool for decision makers.

Keywords: EDD Dispatching Rule, Modified Weighted Due Dates (MWDD), Single Machine Scheduling, Stochastic Processing Time, Weighted Tardiness

1. Introduction

Scheduling concerns the allocation of machines or equipments to a given set of tasks or jobs around time. For single machines there is only one resource or machine available for processing of jobs or tasks. The Single-Machine Total Weighted Tardiness Scheduling Problem (SMTWTSP) is one of the biggest researched problem in the field of Just In Time (JIT) production scheduling. In JIT production scheduling, we effort that jobs are to be finished as their due dates or near due dates for evading the tardiness as well as earliness detriment. If jobs are dealt before its due dates are called early jobs and if jobs are dealt after its due dates are called tardy jobs[1]. The production companies are detriment with great extent if they have early jobs or tardy jobs. Hence, for obtaining absolute scheduled, jobs should be finished exactly on their due dates. The consequences of the tardy delivery of a product, like the loss of reputation between the customers are analyzed[2]. In such surrounding earliness and tardiness are significant. In this paper, we used tardiness based objective function and completion time based objective functions.

The Total Weighted Due Date Tardiness Scheduling Problem (TWDDTSP) is the colligation of the Total Due Date Tardiness Scheduling Problem (TDDTSP). Lateness' $L_i = [\{(C]_i - d_i)\}$ either positive or negative. Tardiness $T_i = \{\max[(C]_i - d_i, 0)\}$ is the positive part of

* *Author for correspondence*

lateness and negative part of lateness is called earliness, $E_i = \{\max[(0, d]_i - C_i)\}$. In our paper, we focus the tardiness of jobs with their weights.

This paper focus on Total Weighted Tardiness Scheduling problem related to Single Machine with no ready times, which can be characterized as $\{1 \| d_i(i) \| \sum W_{ti}\}$[3]. Furthermore, it is also presumed that all jobs processing times are not known in advance. It means processing times of jobs are stochastic, not deterministic in nature. The weights of the jobs are also joined with them to indicate the cognate momentous of jobs. Our goal is to obtain the near optimal or optimal scheduled of job sequence so as to optimize the objective function.

In this particular paper, we introduced two proposed heuristic algorithms that solved n jobs single machine $(n \geq 1)$ Total Weighted Tardiness Scheduling Problem. In the first proposed heuristic algorithm we find an initial sequence using EDD (Earliest Due Dates) rule sometimes it is called Jackson's rule due to R[4]. Jackson, who studied it in 1954 and in the second heuristic algorithm we improved to this solution (Initial solution) using new MWDD (Modified Weighted Due Dates) technique. The performances of these two heuristics are justified by numerical illustration.

The occurrence of the problem related to single machine has the content of the comprehensive research, till now the work of[4]. Total Weighted Tardiness Scheduling Problem is a generalization of tardiness scheduling problems. [5]His works related to theoretical development of Total Tardiness problem is always satisfactory. He extended his results for total weighted tardiness problems and, for the identical problem he developed an effective algorithm. They[6] attained some modest elongation to[5] results and introduced an approach to B&B algorithm. In[7] studied that the special case of weighted tardiness problem can be seen to be Non Polynomial. Due to ramification of SMTWTP, He[8] proved that the minimization of the value of the total weighted tardiness and earliness cost is powerfully non-polynomial. In[9] also shows that total weighted tardiness problem related to single machine scheduling is strongly Non Polynomial. He[10] studied to EDD (Earliest Due Date) rule that it will be optimal, if there is only one tardy job is developed by the Earliest Due Date scheduling of sequence. The scheduling problems with common due date assignment are extensively surveyed by[11,12]. In[10] considered the weighted and unweighted tardiness problem and proposed an algorithm for applying the Modified Due Date (MDD). Firstly[13], proposed the heuristic algorithm that attempted to apply the kinship within the due dates and variables of jobs. The weighted tardiness problem was brilliantly studied by[14]. In[15] introduced a heuristic algorithm for single machine scheduling problem to modify the due date for minimization of unweighted tardiness as well as total tardiness.

He[16] developed a model for minimization of total weighted tardiness as well as total tardiness. They[17] generated an effective method for a Modified Due Date with Weighted and equated it against some other methods for tardiness weighted scheduling. In[18] introduced the Meta heuristic approach compared to heuristics. An algorithm based on Simulated Annealing (SA) was developed by[19] for just-in-time scheduling. They proposed a Genetic Algorithm (GA) meta heuristic algorithm (Memetic Algorithm), for scheduling problems with due dates related to single machine. The later memetic algorithm was further studied by[20] to solve the problems based on weighted tardiness scheduling for the single machine. Processing times of the jobs are to be presumed as stochastic, have been addressed mainly since the 1980's. In[21] studied the Stochastic Scheduling and proposed an algorithm based on weighted tardiness criteria. He[22] delivered the scheduling problem for minimization of the weighted total completion time.

2. Problem Description

In our problem, we studied the Total Weighted Tardiness Scheduling Problem for single machine. In this problem n jobs are to be processed on single machine M with due dates (d_i) and processing time $[(m]_i)$. Processing times are connected to their probabilities $[(p]_i)$ such that $\{0 \leq p_i \geq 1, \sum_{i=1}^{n} [p_i = 1\}]$. The weights $[(w]_i)$ of the jobs are also given to show the priority of jobs. Our goal is to obtain the optimal or near optimal sequence for minimization of the objective functions. The problem is denoted as $\{1 \| d_i \| \sum_{i=1}^{n} [W_{ti}\}]$

2.1 Notations and Parameters Used

- $M =$ Machine
- $p_i =$ Probabilities of i jobs on machine M Where $(i = 1 \text{ to } n)$
- m_i = Processing times of i jobs on M Machine

- d_i = Due Dates of i jobs
- W_{ti} = Weighted Tardiness of i jobs
- w_i = Weight of i jobs
- M_1' = Fictitious Machine
- C_i = Completion Time of i jobs
- $\overline{C_i}$ = Mean Completion Time
- $\sum_{i=1}^{n} C_i$ = **Total Completion time of** or Makespan.
- m_i' = Processing times of i jobs on M_1' Machine
- d_{mi} = Modified Due Dates of i jobs
- $\sum_{i=1}^{n} T_i$ = **Total Tardiness of**
- $\overline{T_i}$ = Mean Tardiness
- $\sum_{i=1}^{n} W_{ti}$ = Total Weighted Tradiness
- $\overline{W_{ti}}$ = Mean Weighted Tardiness
- $W_{ti(max)}$ = Maximum Weighted Tardiness
- N_T = No of Tardy Jobs

3.2 Performance Measures

In this paper we classified performance measures in two different categories, namely:

3.2.1 Completion Time Based Measures

- Total Completion Time or Total Flow Time or Makespan ($\sum_{i=1}^{n} C_i$)
- Mean or Average Completion Time $(\overline{C_i})$ = $\frac{(\sum_{i=1}^{n} C_i)}{n(Total\ no\ of\ jobs)}$

3.2.2 Tardiness Based Measures

- Total Tardiness $\left(\sum_{i=1}^{n} T_i\right)$
- Mean or Average Tardiness $(\overline{T_i})$ = $\frac{(\sum_{i=1}^{n} T_i)}{n(Total\ no\ of\ jobs)}$
- Total Weighted Tardiness $\left(\sum_{i=1}^{n} W_{ti}\right)$
- Mean or Average Weighted Tardiness $(\overline{W_{ti}})$ = $\frac{(\sum_{i=1}^{n} W_{ti})}{n(Total\ no\ of\ jobs)}$
- Total Number of Tardy Jobs (N_T) = $\sum_{i=1}^{n} \delta(T_i)$.
- Where, $\eta(y) = 1\ if\ y > 0\ otherwise\ \eta(y) = 0$

3.3 Assumptions

- All the jobs and machine are available at time Zero.
- The problem comprise of a fixed jobs set which are all completed. The sequence of the jobs does not change. (Static Scheduling problem).
- Processing times of jobs are associated with probabilities. $\left\{\sum_{i=1}^{n} p_i = 1, 0 \le p_i \ge 1\right\}$
- Release time of jobs $r_j = 0$.
- The machine cannot operate more than one operation at a time.
- The preemption is not taken into account. When jobs are began to process on the single machine, it should be finished earlier some other jobs start to process on that particular machine.
- The machine is assumed to be continuously available and Machine breakdowns or maintenance tasks are not considered.
- Setup times are included in the processing time.

3.4 Objective Functions

Objectives Functions

$$
\begin{cases}
\min\left\{\sum_{i=0}^{n} T_i = \text{Total Tardiness}\right\} \\
\min\{\overline{T_i} = \text{Average Tardiness}\} \\
\min\{\sum_{i=1}^{n} W_{ti} = Total\ weighted\ Tardiness \\
\min\{W_{ti(max)} = \text{maximum weighted tardiness}\} \\
\min\{N_T = \text{Number of Tardy Jobs}\} \\
\min\left\{\sum_{i=0}^{n} C_i = \text{Makespan}\right\} \\
\min\{\overline{C_i} = \text{mean completion time}\} \\
\min\{\overline{W_{ti}} = \text{weighted tardiness}\}
\end{cases}
$$

3.5 Mathematical Model for Weighted Tardiness Problem of Single Stage (Machine)

In this model jobs have priorities (weights) attached to their tags which are specified by the term " $w_{(i)}$ ". Consider a set of n jobs (j_1, j_2, j_3, j_4, ...,j_n)with processing time associated with probabilities $(p_1, p_2, p_3 \dots \dots p_n)$, due dates $(d_1, d_2, d_3, d_4, d_5, \dots \dots d_n)$ and weights $(w_1, w_2, w_3, w_4, \dots \dots w_n)$ respectively are processed on single machine M. This model in a matrix form is represented in Table 1.

Table 1. Mathematical Model for Single Machine Weighted Tardiness Problem

Jobs (i)	Machine (M) Processing time (m_i)	Probabilities of processing time (p_i)	Due Dates (d_i)	Weight of Jobs (w_i)
j_1	m_1	p_1	d_1	w_1
j_2	m_2	p_2	d_2	w_2
j_3	m_3	p_3	d_3	w_3
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
j_n	m_n	p_n	d_n	w_n

4. Heuristic Algorithm for Total Weighted Due Dates Tardiness Scheduling (TWDDTS) Problem

Step 1: First we calculate expected processing time $m_i' = m_i * p_i$ about machine M and introduced the new fictitious machine (M_1') with processing time m_i' and reduced the problem with new processing time (m_i').

Step 2: Sequenced the jobs using Earliest Due Date rule (EDD).

According to this technique, jobs are ordered in Shortest Due Dates (SDD) or non-decreasing order of their due dates such that;

$d_1 \le d_2 \le d_3 \le d_4 \le d_5 \dots \dots d_i \le d_{i+1} \le \dots \dots d_n$

Step 3: Now we calculate the tardiness T_i of each job,

$T_i = \{\max(C_i - d_i, 0)\}$

Step 4: Now we calculate weighted tardiness (W_{ti}),

$$W_{ti} = w_i * T_i$$

Step 5: Compute the Total Weighted Tardiness, Average Weighted Tardiness, Total Tardiness, Average Tardiness, Number of Tardy Jobs and Total Completion Time, Average Completion Time.

5. Improved Heuristic Algorithm for Modifed Total Weighted Due Date Tardineess Scheduling (MTWDDTS)

Step 1: First we calculate expected processing time $m_i' = m_i * p_i$ about machine M and introduced the new fictitious machine (M_1') with processing time m_i' and reduced the problem with new processing time (m_i').

Step 2: In this step we modify the due dates. Let d_{mi} be represented the modify due date.

$$d_{mi} = \frac{d_i}{w_i}$$

Step 3: Sequenced the jobs using EDD rule (Earliest Due Date) using the new modified due dates.

Step 4: Now we calculate the tardiness T_i of each job,

$T_i = \{\max(C_i - d_i, 0)\}$

Step 5: Now we calculate weighted tardiness (W_{ti}),

$W_{ti} = w_i * T_i$

Step 6: Compute the Total Weighted Tardiness, Average Weighted Tardiness, Total Tardiness, Average Tardiness, Number of Tardy Jobs and Total Completion Time, average Completion Time,

6. Numerical Illustrations

Consider, 4 jobs are carried out on single Machine **(M)** with their processing time associated with probabilities, due dates and their weights are given in Table 2.

Table 2. Four jobs single machine scheduling problem in matrix form

Jobs (i)	Machine (M) Processing time (m_i)	Probabilities of processing time (p_i)	Due Dates (d_i)	Weight of Jobs (w_i)
1	30	0.4	16	4
2	40	0.2	26	5
3	50	0.3	25	3
4	90	0.1	27	5

6.1 Numerical Solved by Heuristic Algorithm for TWDDTS

As per Step 1: The expected processing times m_i' on machine $(M)_1'$ are as in Table 3.

$\{m_i' \; (m_i' = m_i * p_i)\}$

As per Step 2: New reduce problem is represented in Table 4 using Step 2 to Step 5 and calculate the objective functions for TWDDTS.

Step 3: Result of Objective Function for TWDDTS is shown in Table 5.

Table 3. Fictitious machine for TWDDTS

Jobs (i)	Machine $(M)_1'$ Expected Processing time m_i' $(m_i' = m_i * p_i)$	Due Dates (d_i)	Weight of Jobs (w_i)
1	12	16	4
2	8	26	5
3	15	25	3
4	9	27	5

Table 5. Result of objective functions for TWDDTS

Result of Objective Functions for TWDDTS	
No of Tardy Jobs $N_T = 3$ $\sum_{i=1}^{4}(T_i) = 28$ Mean Tardiness = 7 Total Weighted Tardiness =136	Mean weighted tardiness = 34 Makespan = 118 Mean Completion Time = 29.5

6.2 Numerical Solved by Heuristic Algorithm for MWDDTS

As per Step 1: The expecting processing times on the machine **(M)** are shown in Table 6.

As per Step 2: Modify the due dates ($d_{mi} = \frac{d_i}{w_i}$) and it is represented in the Table 7.

As per Step 3: Sequenced the jobs by applying EDD rule on new modify due dates and new reduced problem is shown in Table 8.

As per Step 3 and Step 4: Calculate the tardiness and weighted tardiness of jobs in Table 9.

Table 10 shows the result of objective functions for MTWDDTS.

Table 6. Fictitious machine for MWDDTS

Jobs (i)	Machine $(M)_1'$ Processing time m_i' $(m_i' = m_i * p_i)$	Due Dates (d_i)	Weight of Jobs (w_i)
1	12	16	4
2	8	26	5
3	15	25	3
4	9	27	5

Table 7. For modify due dates

Jobs (i)	Machine $(M)_1'$ Processing time m_i' $(m_i' = m_i * p_i)$	Due Dates (d_i)	Weight of Jobs (w_i)	$d_{mi} = \frac{d_i}{w_i}$
1	12	16	4	4
2	8	26	5	5.2
3	15	25	3	8.33
4	9	27	5	5.4

Table 4. Calculate the objective functions for TWDDTS

Jobs (i)	Machine $(M)_1'$ $m_i' = m_i * p_i$	Due Dates (d_i)	Completion time (C_i)	Tardiness $T_i = max((C)_i - d_i, 0)$	Weight of Jobs (w_i)	Weighted Tardiness $W_{ti} = w_i * T_i$
1	12	16	12	0	4	0
3	15	25	27	2	3	6
2	8	26	35	9	5	45
4	9	27	44	17	5	85
			$\sum_{i=1}^{4} C_i = 44$	$\sum_{i=1}^{4}(T_i) = 28$		$\sum_{i=1}^{4} W_{ti} = 136$

Table 8. Applying EDD rule

Jobs (i)	Machine (M'_1) Processing time m'_i ($m_i * p_i$)	Due Dates (d_i)	Weight of Jobs (w_i)
1	12	16	4
2	8	26	5
4	9	27	5
3	15	25	3

Table 10. Result of objective functions for MTWDDTS

Result of Objective Functions for MTWDDTS	
No of Tardy Jobs $N_T = 2$ Total Tardiness $\sum_{i=1}^{4}(T_i) = 21$ Mean Tardiness = 5.3 Total Weighted Tardiness =67	Mean weighted tardiness = 16.8 Makespan = 105 Mean Completion Time = 26.3

Table 9. Calculate the objective functions for MTWDDTS

Jobs (i)	Machine (M'_1) $m'_i = m_i * p_i$	Due Dates (d_i)	Completion time (C_i)	Tardiness $T_i = max(C_i - d_i, 0)$	Weight of Jobs (w_i)	Weighted Tardiness $W_{ti} = w_i * T_i$
1	12	16	12	0	4	0
2	8	26	20	0	5	0
4	9	27	29	2	5	10
3	15	25	44	19	3	57
			$\sum_{i=1}^{4} C_i = 44$	$\sum_{i=1}^{4}(T_i) = 21$		$\sum_{i=1}^{4} W_{ti} = 67$

7. Comparative Studies between TWDDTS Heuristic and MTWDDTS Improved Heuristic are Shown in Table 11.

7.1 Remarks

If we solve the same problem by TWDDTS and MTWDDTS heuristic algorithms. The comparatative result in Table 11 show that MTWDDTS heuristic outperform as compared to TWDDTS heuristic. MTWDDTS heuristic minimized all the objective functions as compared to TWDDTS heuristic.

8. Conclusion and Future Research

Just In Time (JIT) production scheduling has received significant attention in few decades. In this paper, two heuristic algorithms are introduced for scheduling problem of total weighted tardiness on the single machine, in which processing times are associated with probabilities to minimize the objective functions $\left(\overline{W_{ti}}, \overline{C_i}, \overline{T_i}, N_T, W_{ti(max)}, \sum_{i=1}^{n} W_{ti}, \sum_{i=1}^{n} T_i, \sum_{i=1}^{n} C_i\right)$.

First, we developed a heuristic algorithm for Weighted Due Date Tardiness Scheduling (TWDDTS) then we developed new improved heuristic algorithm by modifying the due dates. We modified the due dates with weights of jobs. We solved the same numerical by both the heuristic algorithm for justifying to these heuristic algorithms and comparatative result shows that improved heuristic algorithms for MTWDDTS give better results than a heuristic algorithm for TWDDTS. By this improved heuristic algorithm for MTWDDTS we not only minimize the average weighted tardiness, average tardiness, number of tardy jobs but also minimize the makespan and mean completion time. So we concluded that our improved heuristic algorithm outperform.

For future research, this study may further extend by considering various parameters like setup times, breakdown effect, tardiness and earliness cost, etc.

Table 11. Comparative studies between TWDDTS heuristic and MTWDDTS improved heuristic

	TWDDTS Heuristic	MTWDDTS Heuristic
Makespan $\left(\sum_{i=1}^{4} C_i\right)$	118	105
Mean Completion Time $\overline{(C_i)}$	29.5	26.3
Total Tardiness $\left(\sum_{i=1}^{4} T_i\right)$	28	21
Mean Tardiness $\overline{(T_i)}$	7	5.3
No of Tardy Jobs $[(N]_T)$	3	2
Total Weighted Tardiness $\left(\sum_{i=1}^{4} W_{ti}\right)$	136	67
Mean Weighted Tardiness $\overline{(W_{ti})}$	34	16.8
Maximum Weighted Tardiness $[(W]_{ti(max)})$	85	57

meta heuristics (like Genetic Algorithm, Ant Colony Optimization, Simulated Annealing, Tabu Search etc.) approach is also used for obtaining the solution to solve the scheduling problems.

9. References

1. Tyagi N, Abedi M, Varshney RG. A preemptive scheduling and due date assignment for single-machine in batch delivery system. Proceeding of CACCS; India. 2013. p. 60–4.
2. Davis JS, Kanet JJ. Single-machine scheduling with early and tardy completion costs. Nav Res Logi. 1993; 40(1):85–101.
3. Graham RL, Lawler EL, Lenstra JK, Rinnooy Kan AHG. Optimization and approximating in deterministic sequencing and scheduling: A survey. Ann Discrete Math. 1979; 4:287–326.
4. Jackson JR. Scheduling a production to minimize maximum tardiness. Research Report 43. University of California at Los Angeles: Manag Sci Res Project; 1955.
5. Emmons H. One machine sequencing to minimize certain functions of job tardiness. Oper Res. 1955; 17:701–15.
6. Rinnooy Kan AHG, Lageweg BJ, Lenstra JK. Minimizing total costs in one machine scheduling. Oper Res. 1975; 23:908–27.
7. Karp RM. Reducibility among combinatorial problems. Complexity Computers Computations. New York: Plenum Press; 1972. p. 83–103.
8. Lenstra JK, Rinnooy Kan AHG, Brucker P. Complexity of machine scheduling problems. Annals of Discrete Mathematics. 1977; 1:343–62.
9. Lawler EL. A pseudopolynomial algorithm for sequencing jobs to minimize total tardiness. Annals of Discrete Mathematics. 1977; 1:331–42.
10. Baker KR. Introduction to Sequencing and Scheduling. New York: Wiley; 1974.
11. Baker KR, Scudder GD. Sequencing with earliness and tardiness penalties - A review. Oper Res. 1990; 38 (1):22–36.
12. Gordon V, Proth JM, Chu CB. A survey of the state-of-the-art of common due date assignment and scheduling research. Eur J of Oper Res. 2002; 139(1):1–25.
13. Wilkerson LJ, Irwin JD. An improved algorithm for scheduling independent tasks. AIIE Transactions. 1971; 3:239–45.
14. Abdul-Razaq TS, Potts CN, Van WLN. A survey of algorithms for the Single Machine Total Weighted Tardiness Problem. Discrete Appl Math. 1990; 26:235–53.
15. Alidaee B, Rosa D. Scheduling parallel machines to minimize total weighted and unweighted tardiness. Comp and Oper Res.1997; 24:775–88.
16. Panneerselvam R. Modeling the single machine scheduling problem to minimize total tardiness and weighted total tardiness. Int J Manage Syst. 1991; 7(1):37–48.
17. Kanet J, Li X. A weighted modified due date rule for sequencing to minimize weighted tardiness. J of Scheduling. 2004; 7:261–76.
18. Franca PM, Mendes A, Moscato P. A memetic algorithm for

the Total Tardiness Single Machine Scheduling Problem. Eur J of Oper Res. 2001; 132(1):224–42.

19. Vahid MD. A simulated annealing algorithm for JIT Single Machine Scheduling with Preemption and Machine Idle Time. Indian J of Sci and Tech. 2011; 4(5):502–8
20. Maheswaran R, Ponnambalam SG, Aravindan C. A meta-heuristic approach to single machine scheduling problems. The Int J of Adv Manufacturing Techn. 2005; 25:772–6.
21. Forst FG. Bicriterion stochastic scheduling on one or more machines. Eur J of Oper Res. 1995; 80:404–9.
22. Belouadah H, Posner M, Potts C. Scheduling with release dates on a Single Machine to Minimize Total Weighted Completion Time. Discrete Appl Math. 1992; 36:213–31.