Istanbul Commerce University Journal of Science
İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 16(31), Bahar 2017
http://dergipark.gov.tr/ticaretfbd

ISTANBUL COMMERCE UNIVERSITY

Research Article

A BRANCH AND BOUND APPROACH FOR SINGLE MACHINE SCHEDULING PROBLEM*

Sebrina Tadesse DAWD[1] **Berk AYVAZ[2]**

[1]sebry3@yahoo.com
[2]İstanbul Ticaret Üniversitesi, Mühendislik ve Tasarım Fakültesi,
Küçükyalı, 34840 İstanbul
bayvaz@ticaret.edu.tr orcid.org/0000-0002-8098-3611

Abstract

Last decades, scheduling problems have attracted researchers because of the fact that they play a critical role in production planning. This paper studies to minimize the sum weight of lateness on a single machine scheduling problem. There are given n jobs and for each job we have a release date, a processing time, a due date and weight in a constraint working environment. Single machine models are important for various reasons because of the fact that it not only provides insights into the single machine environment but also bottleneck problem. There are various exact methods in order to solve single machine scheduling problem with make span objective function. However, if the objective functions is tardiness, lateness, weighted tardiness, weighted lateness etc. to find exact solution is very difficult. In this paper, branch and bound method is proposed to solve single machine scheduling problem with the total weighed lateness objective for small number of job. The proposed method has applied on a job size of 4, 5 and 8 and provides optimal result.

Keywords: Scheduling, Single Machine, Weighted Total Lateness Minimization, Branch & Bound

Araştırma Makalesi

TEK MAKİNE ÇİZELGELEME PROBLEMİ İÇİN DAL SINIR YAKLAŞIMI

Öz

Son yıllarda çizelgeleme problemleri üretim planlamada kritik bir rol oynadığı için araştırmacıların ilgisini çekmektedir. Bu çalışmada toplam ağırlıklı gecikme süresi minimizasyonu amaçlı tek makine çizelgeleme problemi ele alınmıştır. Verilen n iş için işlerin geliş süresi, müşteriye teslim süresi, işlem süreleri ve iş çevresinin kısıtlarından kaynaklanan işlerin ağırlıkları verilmiştir. Tek makine modelleri sadece tek makine ortamı için bir bakış açısı kazandırmasından değil aynı zamanda darboğaz problemlerinin çözümü için de bir bakış sağladığı için önemlidir. Toplam tamamlanma süresi minimizasyonu için tek makine çizelgeleme problemlerini çözmek için tam çözüm veren birçok metot vardır. Bununla birlikte, gecikme, erken bitirme, ağırlıklı gecikme amaçları söz konusu olduğunda tam çözüm bulmak çok zordur. Bu çalışmada az sayıda iş içeren, toplam ağırlıklı gecikme minimizasyonu problem için dal-sınır algoritması önerilmiştir. Önerilen model 4, 5 ve 8 adet iş için gerçek hayat verileri kullanılarak uygulanmış ve en uygun sonuç alınmıştır.

Anahtar Kelimeler: Çizelgeleme, tek makine, toplam ağırlıklı gecikme minimizasyonu, dal & sınır.

*Received / Geliş tarihi: 22/06/2017 Accepted / Kabul tarihi: 28/09/2017
[2]Corresponding Author/ Sorumlu Yazar : bayvaz@ticaret.edu.tr

1. INTRODUCTION

Scheduling problems have attracted researchers for decades because of the fact that they play a significance role in production planning process. In this study, we consider the single machine scheduling problem, in which the objective function is to minimize the total sum of lateness. The problem can be stated as follows: a non-preemptive single machine scheduling problem with n jobs, associated with each job j, (j = 1,…,n) have several parameters: p_j; the processing time for job j, r_j ; the ready time or release date for job j, [d_{jmin}, d_{jmax}] ; the minimum and maximum due date of job j; $w_{j;}$ weight of the jobs with a constraint working shift hours. The objective is to minimize the sum weight of lateness of all jobs. The single machine weighted tardiness/lateness problem is known to be NP-hard (Yurtkuran and Emel, 2016).

There are lots of studies in the literature related to the single machine scheduling problem some of these studies as following:

Gordon et al. (1997) considered a single machine preemptive scheduling problem to minimize the weighted number of late jobs. Yang et al. (2002) investigated a single machine scheduling problem with a flexible maintenance to minimize the makespan. They assumed that the machine should be stopped to maintain or to reset for a constant time r during the scheduling period. A heuristic algorithm with computational experiments is presented for the addressed problem. Chang et al. (2004) considered a single-machine scheduling problem with release times to minimize the total weighted completion time. They proposed two new heuristics to solve problem. Gupta and Chantaravarapan (2008) take into consideration the single machine scheduling problem with independent family setup times where jobs in each family are processed together. The objective is to minimize total tardiness. A mixed-integer linear programming model can solve small-sized problems but due to the NP-hard nature of the problem, two-phase heuristics including simulated annealing algorithms are proposed to find near-optimal schedules. Atan and Aktürk (2008) focused the single CNC machine scheduling problem with controllable processing times to maximize the total profit that is composed of the revenue generated by the set of scheduled jobs minus the sum of total weighted earliness and weighted tardiness, tooling and machining costs. Batun and Azizoğlu (2009) considered the single machine total flow time problem in which the jobs are non-resumable and the machine is subject to preventive maintenance activities of known starting times and durations. They proposed a branch-and-bound algorithm for the problem. Benmansour et al. (2012) addressed a stochastic single machine scheduling problem for minimizing the expected total weighted deviations of completion times from random common due date. Batsyn et al. (2014) considered the preemptive single machine scheduling problem to minimize the total weighted completion time with arbitrary processing times and release dates. They developed a heuristic method in order to solve problem. Zhang et al. (2014) have proposed two objective functions, which are the weighted sum of the waiting times and the weighted discounted cost function of the waiting times, for single machine. Lee et al. (2014) have considered a single-machine two-agent problem to minimize a weighted combination of the total completion time and the total tardiness of jobs from the first agent given that no tardy jobs are allowed for the second agent. A branch-and bound algorithm is used to derive the optimal sequence and two simulated annealing heuristic algorithms are proposed to search for the near-optimal solutions.

In this paper, we present the branch and bound to minimize the total weight of lateness on a single machine scheduling problem. The presented method applied to the jute softening machine scheduling problem in Ethiopia. In jute production industries, if jobs are done early or tardy, deterioration of products or loss of products can be occur. Hence, to solve these problems, it needs a just in time philosophy to produce slight late and not too early jobs.

The paper is organized as follows. In section 2, methodology of the branch and bound as well as insertion of idle time present. In section 3, problem definition and notation are introduced. In section 4, branch and bound algorithm is applied to real life example. In section 5 computational results are shown.

2. METHODOLOGY

There are various methods in order to solve single machine scheduling problem with makespan objective function. However, if the objective functions is tardiness, lateness, weighted tardiness, weighted lateness etc. to find exact solution is very difficult. In this case, branch and bound method is preferred to solve single machine scheduling problem with small number of job. Branch and bound method does not give exact solution with a big number of jobs. To solve big number of job heuristic methods such as genetic algorithm, simulated annealing, tabu search etc. are used. The computational method for solving this problem can be divided in to two steps:

- Sequencing the jobs by branch and bound method
- Scheduling by inserted idle time in the machine - by blocking system

2.1 Branch and Bound method

a. *Branching procedure:* At any node a feasible solution can be partitioned in to different subsets, each corresponds to a descendant node of the search tree using forward sequencing. A simple branching procedure is branch on level i of the search tree by selecting a job to be scheduled on position i, if in a node of the search tree on level i the set of jobs not yet scheduled jobs is denoted by J and the starting time of the jobs from J by t, for position i we only have to consider jobs k with

$$rj < \min_{k \in J} \left\{ \max\left(t, r_k\right) + p_k \right\} ,$$

To decrease the computational effort we can start t_0=min r_j.

b. *Bounding procedure:* As we know practical scheduling problems are not easily solved, because it needs knowledge based system, heuristic algorithms, and integration with other enterprise functions and so on. In this problem, the searching strategies is calculating the lower bound values in the searching space and branch the smallest value. To eliminate a node at each level we use ATCR rule.

ATCR combines WSPT and MS rule:

$$I_j(t, r_j) = \frac{w_j}{p_j} e^{\left(-\frac{\max(d_j - p_j - \max(r_j, t), 0)}{k_1 \bar{p}} * \left(\frac{-\max(rj - t, 0)}{k_2 \bar{p}}\right)\right.}$$

k_1 = due date scaling parameter (look-ahead parameter)
k_2= release time scaling parameter
w_j=the weight of the jobs
r_j= release time of the jobs
t= the starting time of the jobs
$\bar{p}$=is average p_j of the remaining jobs
To determine the scaling parameter and to get a good schedule first considers:
Due date tightness coefficient $\boldsymbol{\tau}$

$$\tau = 1 - \frac{\sum \bar{d_j}}{nC_{\max}}$$

Due Date Range factor R:

$$R = (d_{j\max} - d_{j\min}) / C_{\max}$$

we use the maximum due date values in the job group.

$C_{\max} = \sum_{j=1}^{n} {}_1 p_j + n \bar{r_j}$ over estimated makespan group

The value of K_1 is between τ and R

$$k1 = 4.5 + R, \; for R \leq 0.5$$
$$k2 = 6 - 2R, \; for R > 0.5$$

To determine the release time scaling parameter K_2

Release time severity factor μ

$$\mu = \frac{\bar{r}}{\bar{p}}$$

$$K_2 = \tau / 2\sqrt{\mu}$$

To simplify the computational effort we use the same value of $K_1\bar{P}$ and $K_2\bar{P}$ in different iteration.

If two jobs are equal objective value, we can broke the tie by checking the sensitivity of the job to tardy or early by using its average due date values and then select the sensitive job for the next sequence.

2.2 Insertion idle time

In this study the procedures of idle time insertion are:
1. From the specified sequence, assign the first job in to the first block and scheduling it to complete at its due date. According to Baker and Trietsch (2009) a schedule Z is optimal if the first job in the first block cannot be tardy and the last job in any block cannot be early. In our case if this job does not tardy, start at least by leaving a gap of the strictly tardy jobs processing time; in the job group r_j greater than the shift hours. In our assumption for each job group, the possible shifting time is:
- $\sum p_j < 8$ considers 8 shifting hours
- $\sum p_j \geq 8$ considers 16 working shifting hours
- $\sum p_j \geq 16$ considers 24 shifting hours

2. In the second block, add the adjacent job in the given sequence. In this case it is possible shift the block in both directions; either shifts close to its possible starting time or far from it, to minimize the objective function within a given processing time.
3. If job j is too early when added to the existing block, then it is rescheduled to complete close to its due date, thus starting a new block. Otherwise job j is added to the existing block, starting when job (j-1) completes. At this stage, if we can achieve a better total cost by shifting all jobs in the block later, we do so. This shift is possible only if we have inserted, or a gap between the previous block and the current block. In addition to this the earliness value of the job is better when it closes to the average due date value. In this time the insertion of idle time between blocks are more advantageous to minimize the earliest value.
4. If the gap between the blocks and the constraint of the possible working shift hour consumed before the block's cost minimized, we merge the blocks. Any further shift now applies to the merged block.
5. Finally, re-schedule the tardy jobs i.e. $r_j \geq the\ total\ shifting\ hrs$ in to the idle time of the machine without interrupting of the scheduled jobs. Because the ready time of each job is no larger than its due date, and then the constraints ensure that jobs start at or after their respective ready times and those jobs does not overlap.

Feasibility and optimality criteria:
- If a job that has a low release time never tardy.
- All jobs have a low release time must be done in one day within a possible shifting hrs.
- The working shift hours are 8, 16 and 24 hours.

3. PROBLEM DEFINITION

In this study, we addresses single machine scheduling problem with the aim of minimizing total weighted lateness time. Some of the assumptions and definitions in this scheduling problem are:

- The release time is the time when the material is ready for softening process
- Machine may be idle, because $r_j \neq 0$, but it never breakdown and are available throughout the scheduling period
- There is permutation
- The due date is the optimal duration of each type of fiber in the bin, and the date of processing in the next production process
- No cancellation – each job must be processed to completion
- The processing times are independent of the schedule and include the setup time
- No machine may process more than one operation at a time
- The technological constraints are known in advance and are unchallengeable
- There is no randomness; because the number of jobs, number of machines, due date and release time is known and fixed
- In advance consider the daily production capacity of softening machine and consider the consumption of each fiber for daily production process as a weight of jobs
- The workers are always available
- Always all raw materials are available in their release time
- The raw material is withdrawn from the store in the FCFS method
- There are adequate number of bin
- There is a limited waiting space
- The possible working hours are somewhat greater than the summation of processing times

From these assumptions and definitions, let the objective function Z(x), assume *x* is the schedule

$$Z(x) = \min \sum_{j=1}^{n} w_j (E_j + T_j)$$

Where,

$$E_j = \max(0, d_{j\max} - (t_j + p_j)) = \max(0, d_{j\max} - C_j)$$

$$T_j = \max(0, (t_j + p_j) - d_{j\max}) = \max(0, C_j - d_{j\max})$$

Where,

$C_j < d_i$

$C_j \leq$ total shift hours

$\sum_{j=1}^{n} p_j \leq \sum_{j=1}^{n} r_j$

$t_j \geq r_j$ because $t_i = r_i + s_i$

$t_i + p_i \leq t_j$ or $p_j + t_j \leq t_i$

E_j : $earliness\ of\ the\ job\ j$
T_j : $Tardiness\ of\ the\ job\ j$
n : Number of jobs
t_j : $Suppose\ a\ job\ j\ can\ be\ start$
C_j : $complition\ time\ of\ jobj$

d_{jmin} - the minimum due date value of the job
d_{jmax} - the maximum due date value of the job
p_j - processing time of the job j
C_j - completion time of job j
s_j - waiting time of job j
w_j – weight of the job j

In a large value of due date and small value of completion time, without any calculation a job j is early. But in our case the earliness value is just defined as a time taken for further treatment in the Bin before passing in to the carding or spreader machine. It must be between minimum and maximum due date values. During the insertion of idle time the earliness value is evaluated as

- $E_j \cong d_{j\max}$ it is too early
- $E_j \cong \overline{d}_{j\max}$ it is better
- $E_j \cong d_{j\min}$ İt is early

Besides this a job j is tardy if:

- $E_j \leq d_{j\min}$ $T_i = d_{j\min} - E_j$
- $rj \geq$ working shift hours
- A job doesn’t completely work in the possible working period.

From the above assumptions and definitions, the problem can be mathematically re-write as: $1/r_j/\sum_{j=1}^{n} w_j(E_j + T_j)$, this problem is NP-hard. Because the simple version of this type of problem; $\sum w_j T_j$ and $\sum w_j E_j$ are NP-hard (Bulbul et al (2007); Tanaka(2012); Baker & Dan, (2009); Du and Leung(1990); Hoogeveen(2005); Lawler(1977); Lenstra et al, (1977).

4. APPLICATION

In this section, we apply the branch and bound method to real life application for the jute softening machine scheduling problem in Ethiopia. As before mentioned, in jute production industries, if jobs are done early or tardy, deterioration of products or loss of products can be occur. Hence, to solve these problems, it needs a just in time philosophy to produce slight late and not too early jobs. The presented method has applied on a job size of 4, 5 and 8 and provides optimal result.

4.1 *For a job size 4 (jute softening machine)*

Table 1 shows the data, which is used in this paper. The job index is related to the practical problem and defined as 1=Alaba; 2= Sidamo; 3 =BWCA; 4=BTD; and all measurements are in hours.

Table 1: Data related to jobs

Job	p_j	r_j	d_{jmin}	d_{jmax}	w_j
1	3.2	32	72	144	36
2	2	4.7	120	144	22
3	2.7	12	216	240	20
4	2.34	11.7	48	96	22

As can be seen in Figure 1, by using the above procedure of the branch and bound we obtain the sequence **{2, 3, 4, 1}** as an optimal solution.

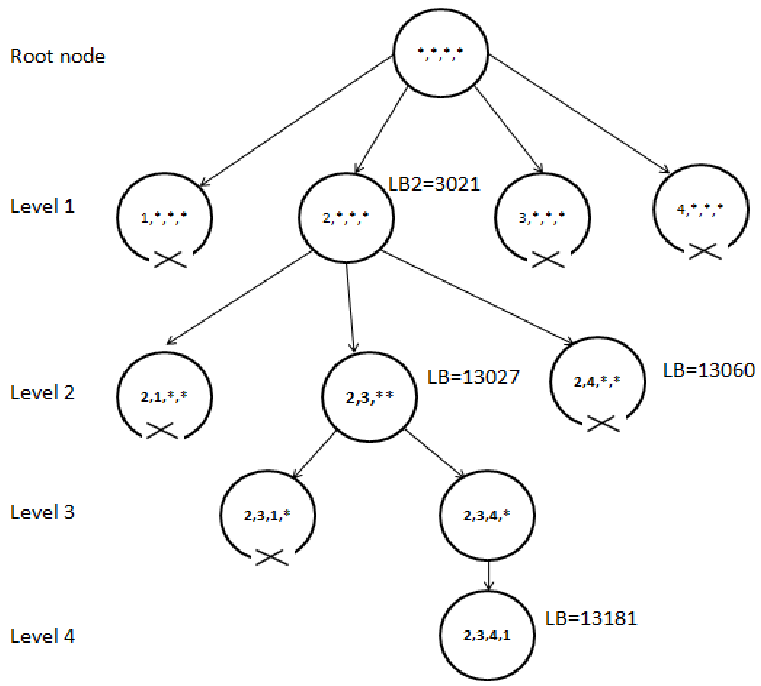

Figure 1. Branch and a Bound schema for 4-job problem final sequence

Table 2. Objective function in branch and bound

job	p_j	t_j	C_j	d_{jmax}	E_j	d_{jmin}	T_j	w_j	Z_j
2	2	4.7	6.7	144	137.3	120	0	22	3021
3	2.7	12	14.7	240	225.3	216	0	20	4506
4	2.34	14.7	17.04	96	78.96	48	0	22	1737
1	3.2	32	35.2	144	108.8	72	0	36	3917
									13181

We obtain optimal solution for objective function by using branch and bound method 13181 as given in Table 2. In addition objective function after idle time insertion is 12875 as seen in Table 3.

Table 3. Objective function after idle time insertion

job	p_j	t_j	C_j	d_{jmax}	E_j	d_{jmin}	T_j	w_j	Z_j
2	2	11.2	13.2	144	130.8	120	0	22	2878
3	2.7	13.2	15.9	240	224.1	216	0	20	4482
4	2.34	21	23.34	96	73	48	0	22	1599
1	3.2	32	35.2	144	108.8	72	0	36	3917
									12875

We can draw Gantt chart as below in Figure 2.

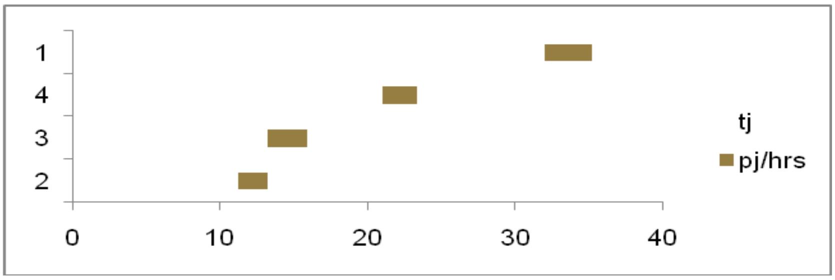

Figure 2. Gantt chart for the final schedule

Note that, Job1 is done in the next day of the scheduling time and the jobs are done within two shifting hours, in other words when a job with a high release time can be done before the scheduled jobs if and only if the machine is idle (see Figure 3).

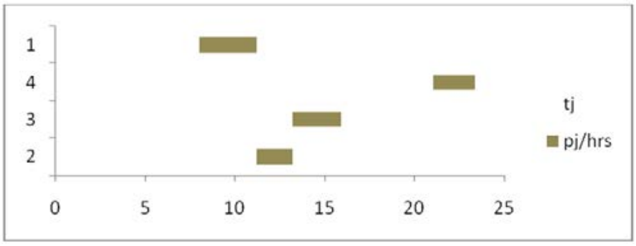

Figure 3. Timetable of the scheduled jobs

4.2 *For a job size 5*

Table 4 shows the data for 5 jobs.

Table 4. Job size 5 scheduling problem

job	p_j	r_j	d_{jmin}	d_{jmax}	w_j
1	2	17	98	144	11
2	2	9	129	212	36
3	3	27	180	203	20
4	3	16	106	173	15
5	2	20	120	186	18

Table 5. Objective function in B&B *{2, 1, 4, 5, 3}*:

job	p_j	t_j	C_j	d_j	w_j	E_j	d_{imin}	w_jE_j	w_jT_j	Z
2	2	9	11	212	36	201	129	7236	0	7236
1	2	17	19	144	11	125	98	1375	0	1375
4	3	16	22	173	15	151	106	2265	0	2265
5	2	20	24	186	18	162	120	2916	0	2916
3	3	27	30	203	20	173	180	3460	140	3600
										17392

We reach optimal solution for objective function by using branch and bound method 17392 as given in Table 5.

Table 6. Objective function after idle time inserted

job	p_j	t_j	C_j	d_j	w_j	E_j	d_{jmin}	w_jE_j	w_jT_j	Z
2	2	15	17	212	36	195	129	7020	0	7020
1	2	17	19	144	11	125	98	1375	0	1375
4	3	16	22	173	15	151	106	2265	0	2265
5	2	20	24	186	18	162	120	2916	0	2916
3	3	27	30	203	20	173	180	3460	140	3600
										17176

In addition objective function after idle time insertion is 17176 as seen in Table 6.

4.3 *For a job size 8*

Table 7 shows the data for 8 jobs.

Table 7. Job size 8 scheduling problem

job	p_j	r_j	d_{jmin}	d_{jmax}	w_j
1	3	27	180	203	15
2	3	11	150	169	6
3	3	21	62	96	15
4	2	5	79	173	4
5	2	8	55	137	19
6	2	7	64	194	16
7	2	16	125	203	14
8	3	10	144	190	11

Table 8. Objective function in B&B {4,6,2,8,7,5,3,1}

job	p_j	t_j	C_j	d_{jmax}	w_j	E_j	d_{jmin}	w_jE_j	w_jT_j	Z
4	2	5	7	173	4	166	79	664	0	664
6	2	7	9	194	16	185	64	2960	0	2960
2	3	11	14	169	6	155	150	930	0	930
8	3	10	17	190	11	173	144	1903	0	1903
7	2	16	19	203	14	184	125	2576	0	2576
5	2	8	21	137	19	116	55	2204	0	2204
3	3	21	24	96	15	72	62	1080	0	1080
1	3	27	30	203	15	173	180	2595	105	2700
										15017

We obtain optimal solution for objective function by using branch and bound method 15017 as given in Table 8. And also objective function after idle time insertion is 14977 as seen in Table 9.

Table 9. Objective function with idle time inserted

job	p_j	t_j	C_j	d_j	w_j	E_j	d_{jmin}	w_jE_j	w_jT_j	Z
4	2	7	9	173	4	164	79	656	0	656
6	2	9	11	194	16	183	64	2928	0	2928
2	3	11	14	169	6	155	150	930	0	930
8	3	14	17	190	11	173	144	1903	0	1903
7	2	17	19	203	14	184	125	2576	0	2576
5	2	19	21	137	19	116	55	2204	0	2204
3	3	21	24	96	15	72	62	1080	0	1080
1	3	27	30	203	15	173	180	2595	105	2700
										14977

5. CONCLUSION

Recent years, scheduling problems take researchers attention because of the fact that they play an important role in production planning process. In this study, we consider the single machine lateness scheduling problem to minimize the total sum of lateness.

In this paper, the branch and bound method is used for all problems. As shown in the above results the insertion of forced or voluntary machine idle time is advantageous to minimize the objective function especially the lateness value. In addition to this, it identifies the working shift, for example in job size 4 and 5 the possible working shifting hours is 16hr/day, and for job size 8, 24hrs/day and also to prove the feasibility of the given sequence in a constraint working environment. The tardy job in all scheduling problems are only 1. However, this job never be solve unless its release time less than the real world working shift hours.
In addition to this all jobs are done without any overlap, this decrease the daily scheduling cost of the planner. In the future we can apply this algorithm for large problems by using meta heuristics algorithms.

REFERENCES

Atan M. O., Akturk M. S., (2008), Single CNC machine scheduling with controllable processing times and multiple due dates, International Journal of Production Research 46, 6087-6111.

Baker K. R. and Trietsch D., (2009), Principles of sequencing and scheduling, A John Wiley& Sons Inc, Hoboken, New Jersey, 2009.

Batsyn M., Goldengorin B., Pardalos M.& Sukhov P., (2014), Online heuristic for the preemptive single machine scheduling problem of minimizing the total weighted completion time, Journal of Optimization Methods and Software, 23(5), 13-17.

Batun S. and Azizoğlu M., (2009), Single machine scheduling with preventive maintenance, International Journal of Production Research, Vol. 47, No. 7, 1753–1771.

Benmansour R., Hamid A.& Artiba A., (2012), Stochastic single machine scheduling with random common due date, Journal International Journal of Production Research, Volume 50, 2012 -Issue 13.

Bülbül K., Kaminsky P., and Yano C., (2007), Preemption in single machine earliness/tardiness scheduling, Springer Science and Business Media, LLC 2007.

Chang P.C, Chung Y.K., Hsieh, J.C., (2004), On single-machine scheduling with release times to minimize total weighted completion time, Journal of the Chinese Institute of Industrial Engineers, Vol. 21, 567-575.

Du J., Leung J.Y.T., (1990), Minimizing total tardiness on one machine is NP-hard; Mathematics of Operations Research Vol. 15, No. 3 (Aug., 1990), 483-495.

Gordon, V., Potapneva E. and Werner F., (1997), Single machine scheduling with deadlines, release and due dates, Optimization, 42, 219-244.

Gupta J.N.D., & Chantaravarapan S., (2008), Single machine group scheduling with family setups to minimize total tardiness, International Journal of Production Research 46, 1707–1722.

Hoogeveen J.A., (2005), Multicriteria scheduling, European Journal of Operational Research Volume 167, Issue 3, 16 December 2005, 592-623.

Lawler E.L., (2005), A “Pseudopolynomial” Algorithm for Sequencing Jobs to Minimize Total Tardiness, Annals of Discrete Mathematics Volume 1, 1977, 331-342.

Lee W., Shiau Y., Chung Y., and Lawson D., (2014), Single-Machine Scheduling to Minimize Total Completion Time and Tardiness with Two Competing Agents, the Scientific World Journal Volume 2014.

Lenstra J.K., Rinnooy A.H.G.K., and Brucker P., (1977), Complexity of machine scheduling problems, Annals of Discrete Mathematics, Vol. 1(1977), 343-362.

Tanaka S., (2012), An Exact Algorithm for the Single-Machine Earliness–Tardiness Scheduling Problem, Just-in-Time Systems, Volume 60 of the series, Springer Optimization and Its Applications, 21-40.

Yang D.L., Hung C.L., Hsu C.J., Chern M.S., (2002), Minimizing the makespan in a single machine scheduling problem with a flexible maintenance, Journal of the Chinese Institute of Industrial Engineers, 19, 63-66.

Yurtkuran A. & Emel E., (2016), A discrete artificial bee colony algorithm for single machine scheduling problems, International Journal of Production Research, 54:22, 6860-6878.

Zhang L., Guan L., Zhou K., (2014), Stochastic Machine Scheduling to Minimize Waiting Time Related Objectives with Emergency Jobs; Discrete Dynamics in Nature and Society, Volume 2014.