DOI: 10.24996/ijs.2024.65.2.28

ISSN: 0067-2904

Solving Maximum Early Jobs Time and Range of Lateness Jobs Times Problem Using Exact and Heuristic Methods

Safanah Faisal Yousif *, Faez Hassan Ali

Mathematics Department , College of Science, Al- Mustansiriyah University, Bagdad, Iraq

Received: 4/1/2023 Accepted: 12/3/2023 Published: 29/2/2024

Abstract

In this paper, a solution to one of the Bicriteria Machine Scheduling Problems (BCMSP) is proposed. This problem focuses on the maximum early jobs time and range of lateness jobs time on a single machine ($1//(E_{max}, R_L)$). First, we derive a subproblem $1//(E_{max} + R_L)$ from the main problem which is a special case for the suggested problem. Secondly, both exact complete enumeration and Branch and Bound (BAB) with two new lower bounds with some heuristic methods to solve the problems are proposed. The results prove the accuracy of BAB to solve the problem for $n \leq 110$ jobs in a reasonable time. In addition, the accuracy of the suggested heuristic methods is compared with the results of the exact methods.

Keywords: Bicriteria Machine Scheduling Problems, Maximum Early jobs Time, Range of Lateness Jobs Times, Branch and Bound method.

حل مسالة القيمة العظمى لوقت الاعمال المبكرة ومدى زمن الاعمال المتأخرة باستخدام الطرق الدقيقة والتقريبية

سفانه فيصل يوسف*, فائز حسن علي

قسم الرياضيات, كلية العلوم ,الجامعة المستنصرية, بغداد, العراق

الخلاصة

في هذا البحث، تم اقتراح حل إحدى مسائل جدولة الماكنة ثنائية المعايير . ومحور هذه المسالة هي القيمة العظمى لوقت الاعمال المبكرة ومدى زمن الاعمال المتأخرة على ماكنة واحدة ($1//(E_{max}, R_L)$). أولاً تم اشتقاق مسالة فرعية من المسالة الرئيسية وهي $1//(E_{max} + R_L)$، ولقد تم اثبات بعض المبرهنات والحالات الخاصة للمسالة المقترحة. وأخيرًا، تم اقتراح بعض الطرق الدقيقة (العد التام وطريقة التفرع والتقيد (BAB) مع قيدين ادنى جديدة) وبعض الطرق التقريبية لحل المسالتين. أثبتت النتائج دقة BAB في حل المسالة لعدد من اعمال $n \leq 110$ في وقت مقبول، ودقة الطرق التقريبية المقترحة مقارنة بنتائج الطرق الدقيقة.

1. Introduction

Scheduling is an essential decision-making practice in many applications such as industrial design, engineering and commercial activities due to the importance of minimizing costs and energy consumption or maximizing profits, performance and efficiency. Scheduling is also used in many manufacturing and service industries. For example, it uses to reduce the cost of production in the industrial operation and to allow companies to be competitive. For that, a good scheduling algorithm can be used, see[1]. The Machine Scheduling Problem (MSP) is

*Email: safanafaysal77@gmail.com

given as follows: There are n jobs given; each job requires one or more operations to be scheduled on one or more machines during a specific time period in order to minimize the given objective function.

Tapan Sen et al. (1988) [2] presented a Branch and Bound (BAB) algorithm to solve the $1//\sum C_j + R_L$ problem. They solved the problem with $n \leq 15$, where a linear combination of the two objectives is considered. The algorithm for minimizing the range of lateness (RL) on a single machine was presented by Liao and Huang which is denoted LH algorithm (1991) [3]. Toktas et al. (2004) [4] solved the problem of minimizing makespan (C_{max}) and maximum earliness (E_{max}) simultaneously in a two-machine flow shop environment. Delphi (2011) [5] proposed an efficient algorithm that can be used to enumerate the set of strict Pareto optimal for the bicriteria scheduling problem without release dates on a single machine which is studied like $1//\sum C_j + E_{max}$. Ibrahim (2014) [6] solved the multicriteria $1//F(T_{max}, E_{max})$ problem and found a possible solution for $1//Lex(T_{max}, E_{max})$ problem. Also, she solved the $1//\mathrm{T_{max}} + \mathrm{E_{max}}$ problem to find an optimal solution or near optimal solution by using the BAB, proposed heuristic algorithm, and local search methods (Descent method, the Tree Type Heuristic Method and Simulated Annealing algorithm), respectively. The authors [7] solved problems $1//(\sum C_j, \sum E_j, E_{max})$ and its special cases problems. Also, local search methods are used for the $1//\sum C_j + \sum E_j + E_{max}$ problem to find near optimum solutions using the proposed algorithm, and local search (Descent and Simulated Annealing) methods. In 2016, S. A. Ali [8] suggested minimizing the problem $1//(\sum C_j, R_L)$, she proposed and applied several exact and approximate algorithms that give an approximate set of efficient solutions for the first time for this problem. Some experimental results are presented to show the applicability of the exact and local search algorithms. With a reasonable amount of time, local search algorithms can solve the problem for up to (900) jobs. In 2021, Ahmed [9] suggested some methods to solve the MCMSP by minimizing $\left(1//(\sum C_j, T_{max}, R_L)\right)$ simultaneously. From the main problem, she deduced the subproblem denoted by $\left(1//(\sum C_j + T_{max} + R_L)\right)$. She also proposed (8) exact, heuristic, and local search methods (Bees Algorithm and Particle Swarm Optimization) to find a set of efficient, approximate, optimal and near optimal solutions for the two problems. She used BAB which solved the main problem without the dominance rule (DR) for $n \leq 18$ and with DR for $n \leq 39$, and solved the subproblem for $n \leq 15$ with the decomposition technique.

In this paper, in section 2, we discuss the MSP concept. In section 3, we introduce the mathematical formulation of the BCMSP for the two suggested problems, namely the maximum earliness (E_{max}) and the range of lateness(R_L). Also, some special cases for problems (ER) are mentioned in section 4. The mathematical formulation of the single-criteria subproblem of (ER) and some special cases for the problems ($ER1$) are introduced in section 5 and section 6, respectively. The DR's for the two problems are introduced in section 7. New Techniques for solving problems (ER) and ($ER1$) are suggested in section 8.

2. Machine Scheduling Problem Concept

This section begins by presenting some important notations, we focus on the performance measures without giving details on the machine environment. It is supposed that there are n jobs, denoted by $1, \ldots, n$, and that these jobs are to be arranged on a collection of machines that are available at all times from time zero onwards and can handle only one job at a time. We only mention the notations that are used for a single machine here. Jobs $j, (j = 1, \ldots, n)$ have [10]:

p_j : The job j has to be processed for a period of length p_j.
d_j : A due date, or the date by which the job should be completed; completion of the job after its due date is permitted, but a penalty is imposed. If the due date must be met without fail, then it is referred to as a deadline, and the common due date is the due date that is the same for all jobs.
s_j: A slack time of job j s.t. $s_j = d_j - p_j$.
C_j : The time at which the processing of job j is completed is called the completion time, such that $C_j = \sum_{k=1}^{j} p_k$.
Now consider we have the sequence σ of jobs then we have:
The earliness $E_j = max\{d_{\sigma(j)} - C_j, 0\}$.
The lateness $L_j = C_j - d_j$.
$R_L = L_{max} - L_{min}$ where $L_{max} = \max_{1\le j\le n}\{L_j\}$, $L_{min} = \min_{1\le j\le n}\{L_j\}$.
$E_{max} = \max_{1\le j\le n}\{E_j\}$.
F is the objective function of ER.
$F1$ is the objective function of EPR.
The following sequencing rules and basic concepts are used in this paper:

Definition (1): The Shortest Processing Time (SPT) rule [11]**:** The problem $1//\sum C_j$ is solved by sequencing all jobs in a non-decreasing order of the processing times (p_j) i.e. $(p_1 \le p_2 \le \cdots \le p_n)$.

Definition (2): The Earliest Due Date (EDD) rule [11]: The problem $1//L_{max}$ is solved by sequencing the jobs in a non-decreasing order of their due dates (d_j) i.e. $(d_1 \le d_2 \le \cdots \le d_n)$.

Definition (3): Minimum Slack Time (MST) rule [9]**:** The problem $1//E_{max}$ is solved by sequencing all jobs in a non- decreasing order of slack time (s_j) i.e. $(s_1 \le s_2 \le \cdots \le s_n)$.

3. Mathematical Formulation of the BCMSP

Let $N = \{1,2,\ldots,n\}$ be a set of jobs that want to be scheduled on a BCMSP with $p_j \le d_j$. The BCMSP can process only one job at one time using the two fields classification, the discussed BCMSP is denoted by $1//(E_{max}, R_L)$. In this paper, the set of efficient solutions that want to be found for the BCMSP can be written for a given schedule $\sigma = (1,2,\ldots,n)$ as follows:

$$\left.\begin{array}{ll} F = min\ (E_{max}, R_L) & \\ \text{s.t.} & \\ C_1 = p_{\sigma_{(1)}} & \\ C_j \ge p_{\sigma_{(j)}}, & j = 1,2,\ldots,n \\ C_j = C_{j-1} + p_{\sigma_{(j)}}, & j = 2,3,\ldots,n \\ L_j = C_j - d_{\sigma_{(j)}}, & j = 1,2,\ldots,n \\ R_{L(\sigma)} = L_{max(\sigma)} - L_{min(\sigma)} & \\ E_j \ge 0, & j = 1,2,\ldots,n \\ E_{max}(\sigma), R_L(\sigma) \ge 0 & \end{array}\right\} \quad (ER)$$

Notice that $E_{max}(\sigma)$ can be solved by MST rule [9]. $R_L(\sigma)$ is an NP-hard problem then BCMSP-ER is also NP-hard.

Remark (1): From definition (2), we always obtain that L_{max}(EDD) $\leq L_{max}$(MST) and from definition (3), we always see that E_{max}(MST) $\leq E_{max}$(EDD).

In the next proposition, we will show the relation between E_{max} and L_{min}.
Proposition (1): $E_{max} = -L_{min}$.
Proof:
$E_{max} = max\{E_j\} = max\left\{max\{-L_j, 0\}\right\}$

$$E_{max} = max\left\{-min\{L_j, 0\}\right\} \quad (1)$$

$min\{L_j, 0\} = min\{L_j\}$ because the minimum of $\{L_j\}$ is always non-positive. Since $min\{L_j\} \leq 0$, then $-min\{L_j\} \geq 0$, then relation (1) can be written as:

$$E_{max} = max\left\{-min\{L_j\}\right\} \quad (2)$$

Since $-min\{L_j\}$ is only one value, it does not need for maximum in relation (2), then relation (2) will be as follows:
$E_{max} = -L_{min}$.
By using the definition of R_L and proposition (1), we obtain:
$R_L = max\{L_j\} - min\{L_j\}$.

$$R_L = L_{max} + E_{max}. \quad (3)$$

Proposition (2): For BCMSP-ER, if $n \to \infty$, then $R_L \to \infty$.
Proof:
Since $R_L = max\{L_j\} - min\{L_j\}$.
If $\to \infty$, $max\{L_j\} \to \infty$. Since $max\{L_j\} \to L_n$ and since $C_j = C_{max} \gg d_n$, we have $L_n \to \infty$ that implies $\lim_{n\to\infty} R_L = L_n - L_{min} \approx L_n$ as $n \to \infty$.
Therefore, $R_L \to \infty$

4. The Most Important Special Cases for BCMSP-ER
Case (4.1): For the BCMSP-ER, If EDD and MST rules are identical, then we obtain an efficient solution.

Proof: Let $\sigma =$ EDD $=$ MST be a sequence that satisfies EDD and MST rules in the same time, σ will minimize E_{max} such that there is no sequence π which satisfies $E_{max(\pi)} < E_{max(\sigma)}$,and σ will minimize R_L that means there is no sequence π which satisfies $R_{L(\pi)} \leq R_{L(\sigma)}$. Therefore, there is no sequence π s. t. $E_{max(\pi)} < E_{max(\sigma)}$ and $R_{L(\pi)} \leq R_{L(\sigma)}$, Then the efficient solution for BCMSP-ER is given by the best sequence σ.

Case (4.2): IF all jobs are late, then the BCMSP-ER is changed to $1//L_{max}$.
Proof: Since all jobs are late, the $E_j = 0 , \forall_j$ then $E_{max} = 0$, then the BCMSP-ER $1//(E_{max} , R_L) = 1//R_L$.
Since $R_L = L_{max} - L_{min}$ and $E_{max} = -L_{min} = 0$.
The BCMSP-ER $1//(E_{max} , R_L)$ will be $1//max\{L_j\}$ and this is the definition of the problem $1//L_{max}$.

Remark (2): It is known that the problem $1//L_{max}$ is solved by EDD rule [12]**,** then the BCMSP-ER can be solved by the EDD rule when all jobs are late.
Case (4.3): For BCMSP-ER if $p = p_j \ \forall_j$, then the problem can be solved by the EDD rule.
Proof: Assume we have the sequence σ**.**
It is known that $L_j = C_j - d_{\sigma(j)}$, and since $p = p_j \, \forall_j$ then $C_j = jp$, then $L_j = jp - d_{\sigma(j)}, E_{max} = -min\{L_j\} = -min\{jp - d_{\sigma(j)}\}$
From Proposition (1), we have
$R_L(\sigma) = max\{L_j\} + E_{max}(\sigma) = max\{jp - d_{\sigma(j)}\} - min\{jp - d_{\sigma(j)}\}$
Now assume that we have the sequence $\sigma =$ EDD rule:
$R_L(\sigma) = max\{L_j\} + K, K = min\{p - d_{\sigma(1)}\} \leq 0$ is constant.
$R_L(\sigma) = max\{p - d_{\sigma(1)}, 2p - d_{\sigma(2)}, \dots, np - d_{\sigma(n)}\} + K.$

This relation depends on the variable $d_{\sigma(j)}$, then $R_L(\sigma)$ and $E_{max}(\sigma)$ depend on $d_{\sigma(j)}$ only, then the EDD rule is found an efficient solution for BCMSP-ER.

Case (4.4): For BCMSP-ER, if $d = d_j \ \forall_j$, then the unique efficient solution for sequence σ is obtained by:

$$(E_{max}, R_L) = (d - p_{\sigma(1)}, C_{max} - p_{\sigma(1)})$$

Where $p_{\sigma(1)}$ is the largest p_j in the schedule σ and no matter for other jobs how they arranged.
Proof: From Proposition (1):

$$E_{max} = -min\{L_j\} = -min\{C_j - d_j\} = -\{min\{C_j\} - d\} = -min\{C_j\} + d = d - C_j$$
$$= d - p_{\sigma(1)}.$$

To minimize E_{max}, $p_{\sigma(1)}$ must be the largest p_j in σ.
$R_L = max\{L_j\} + E_{max} = max\{C_j - d\} + p_{\sigma(1)}$
$R_L = max\{C_j\} - d + d - p_{\sigma(1)} = max\{C_j\} - p_{\sigma(1)} = C_{max} - p_{\sigma(1)},$
To minimize $R_L, p_{\sigma(1)}$ must be the largest p_j in σ.
Therefore, $(E_{max}, R_L) = (d - p_{\sigma(1)}, C_{max} - p_{\sigma(1)})$.

Remark (3): For BCMSP-ER, from case (4.4), notice that there is no effect on the arrangement of other jobs of $\sigma(j),\ j = 2, \dots, n$, so we have $(n-1)!$ sequences that give a uniquely efficient solutions.

Case (4.5): For BCMSP-ER, if there are some jobs with $d_j \geq C_{max}$, then there is a chance to obtain an efficient solution if this (these) job (jobs) is arranged last.

Case (4.6): For BCMSP-ER, if $p_j = p$ and $d_j = d\forall_j$, then there is a uniquely efficient solution with n! sequences with constant objective function:
$(E_{max,} R_L) = (d - p\ , (n-1)p)$ if all jobs are early. $(E_{max}, R_L\ ) = ((0, (n-1)p)$ if all jobs are late.
Proof: It is known that:
$C_j = \sum_{k=1}^{j} p_k = jp$, then $L_j = jp - d$
$L_{max} = max\{L_j\} = np - d$ and $L_{min} = \min\{L_j\} = p - d$

$$\therefore\ R_L = L_{max} - L_{min} = np - d(p - d) = np - p = (n-1)p \qquad (4)$$

$E_j = max\{-L_j, 0\} = max\{d - jp, 0\}$

$$E_j = max\{d - p, d - 2p, \dots, d - jp, \dots, d - np, 0\} \qquad (5)$$

(a). If all jobs are early, this means $d > C_n = np$, then the maximum difference in relation (5) is $d - p$, then

$$E_{max} = d - p \quad (6)$$

From relations (4) and (6), we obtain:
$(E_{max}, R_L) = (d - p , (n - 1)p)$.
(b). If all jobs are late (except the first job), this means $d = p$, then , $E_j = 0\ \forall_j$, then

$$E_{max} = 0 \quad (7)$$

$L_{max} = max\{L_j\} = np - d = np - p = (n - 1)p$ and $L_{min} = p - d = 0$

$$R_L = L_{max} - L_{min} = (n - 1)p - 0 = (n - 1)p \quad (8)$$

Form relations (7) and (8), we obtain:
$(E_{max}, R_L) = (0, (n - 1)p)$.

5. Mathematical Formulation of Single-Criteria Subproblem of BCMSP-ER

In this section, we will discuss subproblem of BCMSP-ER, this problem is bi-objective (BOMSP), which is defined by $1//(E_{max} + R_L)$, and denoted by BOMSP-EPR We will try to find the optimal solution for the BOMSP by using schedule $\sigma = (1,2, \dots , n)$, it can be formulated as:

$$\left.\begin{array}{ll}
F = min\ (E_{max} + R_L) & \\
\text{s.t.} & \\
C_1 = p_{\sigma_{(1)}} & \\
C_j \geq p_{\sigma_{(j)}}, & j = 1,2, \dots , n \\
C_j = C_{j-1} + p_{\sigma_{(j)}}, & j = 2,3, \dots , n \\
L_j = C_j - d_{\sigma_{(j)}}, & j = 1,2, \dots , n \\
R_{L(\sigma)} = L_{max(\sigma)} - L_{min(\sigma)} & \\
E_j \geq 0, & j = 1,2, \dots , n \\
E_{max}(\sigma), R_L(\sigma) \geq 0 &
\end{array}\right\} \quad \dots (EPR)$$

The BOMSP is an NP-hard problem.

Proposition (3): Let σ be the schedule that gives one of the efficient solutions for BCMSP-ER if and only if σ gives the optimal solution for BOMSP-EPR.
Proof:
Let $S = \{\pi_1, \pi_2, \dots , \pi_r\}$ be the set of all efficient schedules which give the efficient solutions $\big((f, g) = (f_i, g_i)\big)\ \forall\ i$ =1,2, … , r for BCMSP-ER. Suppose that the optimal solution $(f_k + g_k)$ for problem $(ER1)$ is not efficient solution for the problem (ER), then the schedule σ which gives $(f_k + g_k)\ \sigma \notin S$, then $f_i + g_i < f_k + g_k\ \ \forall\ i = 1,2, \dots , r$, then $f_k + g_k$ is not the optimal solution for BOMSP-EPR and that is C!.

Let σ be a schedule that gives an optimal solution $f_k + g_k$ for BOMSP-EPR, Suppose σ does not give the efficient solution for problem (ER), this means that $\sigma \notin S$ and that implies there exists an efficient solution $f_i + g_i < f_k + g_k$ that mains $(f_i + g_i)$ is an optimal solution for BOMSP-EPR and its value is less than $f_k + g_k$ that means $f_k + g_k$ is not the optimal solution for BOMSP-EPR and that is C!.

Proposition (4): Let BOMSP-EPR, if $n \to \infty$, then $E_{max} + R_L \to \infty$.
Proof: See proposition (2).

6. The Most Important Special Cases for the BOMSP-EPR

Case (6.1): For BOMSP-EPR, if EDD and MST rules are identical, then the final result is an optimal solution.
Proof: Let $\sigma = \text{EDD} = \text{MST}$ be a sequence that satisfies EDD and MST rules at the same time, then σ will minimize E_{max} such that there is no sequence π that satisfies
$E_{max(\pi)} < E_{max(\sigma)}$, σ will minimize R_L that means there is no sequence π ,
such that $R_{L(\pi)} \le R_{L(\sigma)}$, then we have no π s.t
$E_{max(\pi)} < E_{max(\sigma)}$ and $R_{L(\pi)} \le R_{L(\sigma)}$. Therefore, σ is the best sequence which gives an optimal solution for BOMSP-EPR.

Case (6.2): IF all jobs are late, then BOMSP-EPR is changed to $1//L_{max}$.
Proof: See case (4.2) of BCMSP-ER.

Case (6.3): For the BOMSP-EPR, if $p = p_j\ \forall_j$, then the problem can be solved by the EDD rule.
Proof: Assume we have the sequence σ.
It is know $L_j = C_j - d_{\sigma(j)},$ and since $p = p_j\ ,\forall_j\ then\ C_j = jp,$
then $L_j = jp - d_{\sigma(j)}, E_{max} = -min\{L_j\} = -min\{jp - d_{\sigma(j)}\}$
From Proposition (1), we have
$R_L(\sigma) = max\{L_j\} + E_{max}(\sigma) = max\{jp - d_{\sigma(j)}\} - min\{jp - d_{\sigma(j)}\}$
assume that we have the sequence $\sigma = EDD$ rule:
$R_L(\sigma) = max\{L_j\} + K,$
$K = min\{p - d_{\sigma(1)}\} \le 0$ is constant.
$R_L(\sigma) = max\{p - d_{\sigma(1)}, 2p - d_{\sigma(2)}, \dots, np - d_{\sigma(n)}\} + K$
This relation depends on the variable $d_{\sigma(j)}$, then $R_L(\sigma)$ and $E_{max}(\sigma)$ depend on $d_{\sigma(j)}$ only, then the EDD rule is found an optimal solution for BOMSP-EPR.

Case (6.4): For BOMSP-EPR, if $d = d_j\ \forall_j$, then the unique efficient solution for sequence σ is obtained by:
$E_{max+}R_L = C_{max} + d - 2p_{\sigma(1)}$.
Where $p_{\sigma(1)}$ is the largest processing time in the schedule σ, and no matter for other jobs how they arranged.
Proof: From case (4.4) for the BCMSP-ER, we prove that:
$(E_{max}, R_L) = (d - p_{\sigma(1)}, C_{max} - p_{\sigma(1)})$
Then for BOMSP-EPR, we obtain:
Therefore, $E_{max+}R_L = C_{max} + d - 2p_{\sigma(1)}$.

Remark (4): For BOMSP-EPR, from case (6.6), notice that there is no effect on the arrangement of other jobs of (j), $j = 2, \dots, n$, so we have $(n-1)!$ sequences that give the optimal solution.

Case (6.5): For BOMSP-EPR, if there are some jobs with $d_j \ge C_{max}$, then there is a chance to obtain the optimal solution if this (these) job(s) is (are)arranged last.

Case (6.6): For BOMSP-EPR, if $p_j = p$ and $d_j = d\ \forall_j$, Then there exits an optimal solution with n! sequences with the following constant objective functions:

a. $E_{max} + R_L = d + (n-2)p$ if all jobs are early.
b. $E_{max} + R_L = (n-1)p$ if all jobs are late.

Proof: From case (4.6) of BCMSP-ER:

(a). If all jobs are early, this means we have:
$(E_{max}, R_L) = (d-p, (n-1)p)$.
So for the problem $(ER1)$, we obtain:

$E_{max} + R_L = d + (n-2)p$.

(b). If all jobs are late (except the first job), this means $d = p$, then from relations (7) and (8), then:
$E_{max} + R_L = 0 + (n-1)p = (n-1)p$

Example (1): Table -1 shows the results of applying the cases of BCMSP-ER and BOMSP-EPR using different examples for $n = 4$.

Table 1: Results of applying the cases of BCMSP-ER and BOMSP-EPR for $n = 4$.

Case	p_j and d_j	Conditions	Results		Values
			BCMSP-ER	**BOMSP-EPR**	
1	$p_j = 6\ 1\ 9\ 5$ $d_j = 11\ 13\ 25\ 27$	EDD = MST	ES	OS	(9,4)=13
2	$p_j = 8\ 7\ 2\ 1$ $d_j = 8\ 15\ 17\ 27$	All jobs are late	$1//L_{max}$		(9,9)=18
3	$p_j = 10$, $d_j = 20\ 12\ 26\ 29$	$p = p_j, \forall_j$	solved by EDD rule		(3,14)=17
4	$p_j = 9\ 10\ 2\ 10$ $d_j = 10$,	$d = d_j, \forall_j$	ES $(d - p_{\sigma(1)}, C_{max} - p_{\sigma(1)})$ $p_{\sigma(1)}$ is the largest p_j	OS $C_{max} + d - 2p_{\sigma(1)}$ $p_{\sigma(1)}$ is the largest p_j	(0,21)=21
5	$p_j = 6\ 3\ 5\ 2$ $d_j = 15\ 17\ 12\ 10$	$d_2 \geq C_{max}$	ES if job d_2 arranged last	OS if job d_2 arranged last	(7,6)=13
6	$p_j = 5$, $d_j = 21$	$p = p_j, \forall_j$ $d = d_j, \forall_j$ all jobs are early	$(d-p, (n-1)p)$	$d + (n-2)p$	(16,15)=31
	$p_j = 5$ $d_j = 5$	$p = p_j, \forall_j$ $d = d_j, \forall_j$ all jobs are late (except 1st job)	$(0, (n-1)p)$	$(n-1)p$	(0,15)=15

Where ES and OS are an efficient and optimal solutions respectively.

7. Dominance Rules for BCMSP-ER and BOMSP-EPR

To shorten the current sequence, many dominance rules may be used (DRs). Because DRs typically clarify some (all) parts of the path to achieving an acceptable limit for the objective function, they could be useful when figuring out a node in the BAB procedure that can be dismissed before determining the lower bound (LB). Obviously, DRs are extremely effective when an endpoint may be neglected despite having a suboptimal LB. DRs can be used in the BAB strategy to reduce all nodes that are also controlled by others. As a result of these improvements, the required number of nodes to reach the best solution is significantly reduced [10].

Definition (4) [**13**]**:** If G is a graph with n vertices, then the adjacency matrix of G is the matrix $\boldsymbol{A}(\boldsymbol{G}) = [\boldsymbol{a_{ij}}]$, whose $i^{th}$and j^{th} component are 1 if there exists at least one edge between V_i and V_j and zero otherwise, where:

$$a_{ij} = \begin{cases} 0, & \text{if } i = j \text{ or } j \nrightarrow i, \\ 1, & \text{if } i \rightarrow j, \\ a_{ij} \text{ and } \bar{a}_{ij}, & i \leftrightarrow j. \end{cases}$$

Remark (5) [13]: To see that $i \rightarrow j$ for a problem $1//E_{max}$ if $p_i \leq p_j$ and $s_i \leq s_j$, then we obtain the optimal solution.

Example (2): For the BCMSP-ER, suppose we have the following data for $= 5$:

	1	**2**	**3**	**4**	**5**
$\boldsymbol{p_j}$	7	9	6	8	10
$\boldsymbol{d_j}$	15	20	12	18	17
$\boldsymbol{s_j}$	8	11	6	10	7

By using Remark (4) rule, we obtain the DR's which is mentioned in Figure -1.

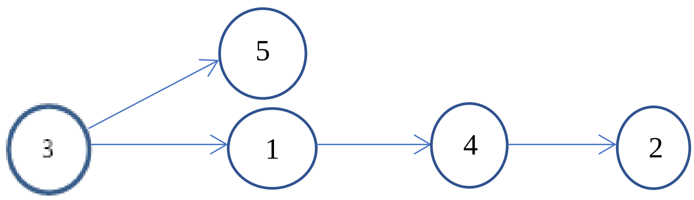

Figure 1: the DRs of example (2).

8. New Solving Techniques for BCMSP-ER and BOMSP-EPR

Throughout this section, we will recommend applying the exact methods such as the complete enumeration method (CEM) and constructing the Branch and Bound method (BAB) as well as suggest new heuristic methods for the two problems.

8.1 Branch and Bound Method for BCMSP-ER and BOMSP-EPR

The BAB method is an exact method for finding an optimal solution to an NP-hard problem. To minimize an objective function for a particular scheduling problem.

Before we discuss the BAB method, we have to define the upper bound (UB) and lower bound (LB) for using BAB to solve the two problems.

For UB, we suggest using the SPT rule, while for LB we use the EDD or MST rule for the unsequenced art, so we denote the BAB by BAB(Rule) technique. In this technique, we use the classical BAB(Rule) to determine a set of the Pareto optimal solutions for BCMSP-ER and the optimal solution for BOMSP-EPR. The BAB (Rule) steps are given as follows:

Algorithm (1): BAB(Rule) Algorithm

Step (1): INPUT n, p_j and d_j for $j = 1,2, \ldots, n$, and Rule="MST" or "EDD".

Step (2): SET $S = \emptyset$, define $F(\sigma) = \big(E_{max}(\sigma), R_L(\sigma)\big)$ for any σ.

Step (3): Determine the S=UB by σ = SPT rule. For this order σ, compute $F(\sigma)$, and set the UB $= F(\sigma)$ at the parent node of the search tree.
Step (4): $i = i + 1$, in level i, for every node in the search tree and for every sequence of jobs δ, compute a $LB(\delta)$ as follows: $LB(\delta)$ = cost of sequence jobs(δ) for the objective functions + cost of unsequenced jobs obtained by sequence the jobs in just one Rule (EDD or MST) in one time.
Step(5): Check that if δ is dominated UB, then $S = S \cup \{\delta\}$, and branch from this node, if its not
Step(6): When finishing each node, check the efficient (optimal) solution S, say S' s.t. $S' \subseteq S$.
Step(7): Change $UB = S'$.
Step(8): GOTO Step(4) until finishing checking all levels ($i = n$).
Step(9): Calculate $O_i = F(\{S'\})$, for $i = 1, \dots, k$ (k number of the efficient point)**.**
Step(10): Stop.

8.2 Heuristic Methods for BCMSP-ER and BOMSP-EPR

The CEM and BAB(Rule) take a long time to obtain an efficient (optimal) solution and have failed to find this solution for big problems. So we should use heuristic methods to find good solutions. In this section, we propose two heuristics for solving BCMSP-ER and BOMSP-EPR.

First Algorithm: MST-SPT-ERL Method

The summary of this method is using MST and calculating the objective function of the two problems, and then putting the second job in the first place and the other jobs still arranged by MST rule and calculating the objective function for the second arrangement, and so on until we obtain n sequences, then we re-use the same technique for SPT rule to obtain n sequences, so in the end, we have $2n$ sequences. We filter them to obtain the most efficient (optimal) solution(s). Algorithm (3) shows the MST-SPT-ERL's steps.

Algorithm (2): **MST-SPT-ERL Method**
Step (1): INPUT n , p_j and d_j for $j = 1,2, \dots, n, \delta = \varphi$.
Step (2): Organize jobs according to MST rule (σ_1), and determine, $F_{11}(\sigma_1); \delta = \delta \cup \{F_{11}(\sigma_1)\}$.
Step (3): The rest $i = 2, \dots, n$, move job i to the first place of σ_{i-1} to gain σ_i and determine $F_{1i}(\sigma_i); \boldsymbol{\delta} = \boldsymbol{\delta} \cup \{F_{1i}(\sigma_i)\}$ **.**
Step (4): Organize jobs according to SPT rule (π_1) ,and determine $F_{21}(\pi_1); \delta = \delta \cup \{F_{21}(\pi_1)\}$.
Step (5): The rest $i = 2, \dots, n$, move job i to the first place of π_{i-1} to gain π_i and determine $F_{2i}(\pi_{1i}); \delta = \delta \cup \{F_{2i}(\pi_i)\}$.
Step (6): Classifier the set δ to gain a collection of efficient (optimal) solution(s) for BCMSP-ER (BOMSP-EPR).
Step (7): OUTPUT The set of efficient solution δ or optimal solution.
Step (8): STOP.

The idea of the second heuristic method is summarized by finding a sequence sort with minimum s_j and p_j which does not contradict DR and calculate the objective function, The main steps of DR-ERL are as follows:

The second heuristic approach is outlined by determining the objective function and discovering an array sort with lower limit s_j and p_j that does not disagree with DR. Algorithm (3) shows the DR-ERL's steps.

Algorithm (3): DR-ERL Heuristic Method
Step (1): INPUT: n, p_j and d_j for $j = 1,2,\dots,n, \delta = \varphi$.
Step (2): Apply remark (4) to determine the matrix A of DR; $N = \{1,2,\dots,n\}, \delta = \varphi$.
Step (3): Determine the lowest σ_1 with minimum p_j which does not contradict matrix A (DR), if there exists more than one job choose arbitrary, $\delta = \delta \cup \{\sigma_1\}$.
Step (4): Determine the lowest σ_2 with minimum s_j which is not contradiction with matrix A (DR), if there exists more than one job choose arbitrary, $\delta = \delta \cup \{\sigma_2\}$.
Step (5): Classifier the set δ.
Step (6): Determine $F(\delta)$.
Step (7): OUTPUT the set of efficient (optimal) solution δ.
Step (8): End.

9. Applying Solving Techniques for BCMSP-ER and BOMSP-EPR

We generate the values of p_j and d_j for all example randomly s.t. $p_j \in [1,10]$ and

$$d_j \in \begin{cases} [1,30], & 1 \le n \le 29. \\ [1,40], & 30 \le n \le 99. \\ [1,50], & 100 \le n \le 999. \\ [1,70], & \text{otherwise}. \end{cases}$$

Under an important condition that $d_j \ge p_j$ [14], for $j = 1,2,\dots,n$.
Now we introduce the following important abbreviations:

Ex	:	Example Number.
Av	:	Average.
AAE	:	Average Absolute Error.
AT/S	:	Average of Time per second.
Av	:	Average.
R	:	$0 <$ Real < 1.
F	:	Objective Function value for BCMSP-ER.
F_1	:	Objective Function value for BOMSP-EPR.

Each example of n is revered for 5 experiments**.**
Table 2 Shows the comparison results between BAB(MST) and BAB(EDD) with CEM for $n = 4{:}11$ for the two problems.

Table 2: The comparison results between BAB(MST) and BAB(EDD) with CEM for $n = 4{:}11$ for the two problems.

n	CEM			BAB(MST)				BAB(EDD)			
	F	$F1$	AT/S	F	$F1$	AT/S	AAE	F	$F1$	AT/S	AAE
4	(9.2,9)	18.2	R	(9.2,9)	18.2	R	0	(9.2,9)	18.2	R	0
5	(5.8,14.2)	20	R	(5.8,14.8)	20.6	R	0.6	(5.8,14.2)	20	R	0
6	(5.6,17.2)	22.8	R	(5.6,17.2)	22.8	R	0	(5.6,17.2)	22.8	R	0
7	(6.4,18.2)	24.6	R	(6.4,18.6)	25	R	0.4	(6.4,18.2)	24.6	R	0
8	(3.6,18.6)	22.2	R	(3.6,19.4)	23	R	0.8	(3.6,18.6)	22.2	R	0
9	(3.4,20)	23.4	6.8	(3.4,20.6)	24	R	0.6	(3.4,20)	23.4	R	0
10	(2.4,33)	35.4	85.5	(2.4,33.2)	35.6	R	1.2	(2.4,33)	35.4	R	0
11	(6,37.4)	43.4	1010.1	(6,37.8)	43.8	R	0.4	(6,37.4)	43.4	R	0
Av	**(5.3,20.9)**	**26.2**	**137.8**	**(5.3,21.3)**	**26.6**	$\boldsymbol{R}$	**0.5**	**(4.9,20.9)**	**26.2**	$\boldsymbol{R}$	**0**

Table 3 shows the comparison results between BAB(MST) with BAB(EDD) for $n = 20:10:110$ for the two problems.

Table 3: Comparison results between BAB(MST) and BAB(EDD) for $n = 20:10:110$ for the two problems.

n	BAB(MST)			BAB(EDD)		
	F	***F1***	***AT/S***	***F***	***F1***	***AT/S***
20	(3.2,89.6)	92.8	*R*	(2.8,89.2)	92	*R*
30	(2.8,127.8)	130.6	2.2	(2.8,127.8)	130.6	2.3
40	(1.6,194.6)	196.2	4.1	(1.6,194.2)	195.8	4.4
50	(2,233.8)	235.8	9.5	(2,233.6)	235.6	9.5
60	(1,294.8)	295.8	19.7	(1,294.8)	295.8	16.5
70	(2,350.4)	352.4	30.4	(2,350.2)	352.2	20.4
80	(1,399)	400	35.7	(1,399)	400	27.6
90	(1.2,469.8)	471	73.4	(1.2,469.6)	470.8	68.6
100	(0.8,505.2)	506	167.8	(0.8,505.2)	506	145.9
110	(1.4,573.8)	575.2	301.1	(1.4,573.6)	575	173.4
Av	**(1.7,323.8)**	**325.5**	**64.39**	**(1.6,323.7)**	**325.3**	**46.8**

Remark (6):

- The BAB(Rule) is solved BCMSP-ER up to $n = 110$ in a reasonable time.
- From tables (2) and (3), we notice that the result of applying BAB(EDD) is closer to CEM than BAB(MST), so it will be used to compare with other solving techniques.

Table -4 shows the comparison results between MST-SPT-ERL and DR-ERL with CEM for $n = 4:11$ for the two problems.

Table 4: Comparison results between MST-SPT-ERL and DR-ERL with CEM for $n = 4:11$ for the two problems.

n	CEM(*F*)			MST-SPT-ERL				DR-ERL			
	F	***F1***	***T/S***	***F***	***F1***	***T/S***	***AAE***	***F***	***F1***	***T/S***	***AAE***
4	(9.2,9)	18.2	R	(9.2,9)	18.2	R	0	(9.8,9.6)	19.4	R	1.2
5	(5.8,14.2)	20	R	(5.8,15)	20.8	R	0.8	(7.8,16.2)	24	R	4
6	(5.6,17.2)	22.8	R	(5.6,18)	23.6	R	0.8	(6.4,18)	24.4	R	1.6
7	(6.4,18.2)	24.6	R	(6.4,19.4)	25.8	R	1.2	(9.6,21.4)	31	R	6.4
8	(3.6,18.6)	22.2	R	(3.6,19.8)	23.4	R	1.2	(8.2,23.2)	31.4	R	9.2
9	(3.4,20)	23.4	6.8	(3.4,21.8)	25.2	R	1.8	(4.4,21)	25.4	R	2
10	(3,33.6)	36.6	74.2	(2.4,33.4)	35.8	R	0.8	(5.4,36)	41.4	R	4.8
11	(6,37.4)	43.4	920.01	(6,38.6)	44.6	R	1.2	(7.2,38.6)	45.8	R	2.4
Av	**(5.3,21.02)**	**26.4**	**125.1**	**(5.3,21.8)**	**27.1**	**R**	**0.9**	**(7.3,23)**	**30.3**	**R**	**3.9**

Table 5 shows the comparison results between MST-SPT-ERL and DR-ERL with BAB(EDD) for $n = 30:20:110$ for the two problems.

Table 5: Comparison results between MST-SPT-ERL and DR-ERL with BAB(EDD) for $n = 30:20:110$ for the two problems.

n	BAB(F,EDD)			MST-SPT-ERL				DR-ERL			
	F	$F1$	T/S	F	$F1$	T/S	AAE	F	$F1$	T/S	AAE
30	(2.8,127.8)	130.6	2.8	(2.8,129.4)	132.2	R	1.6	(4.8,129.8)	134.6	R	4
50	(2,233.6)	235.6	10.4	(2,235.4)	237.4	R	1.8	(5.6,237.2)	242.8	R	7.2
70	(2,350.2)	352.2	19.9	(2,351)	353	R	0.8	(5.2,353.4)	358.6	R	6.4
90	(1.2,469.6)	470.8	70.2	(1.2,470.4)	471.6	R	0.8	(7.8,476.2)	484	R	132
110	(1.4,573.6)	575	170.4	(1.4,574.6)	576	R	1	(6,578.2)	584.2	R	9.2
Av	**(1.8,350.9)**	**352.8**	**54.7**	**(1.8,352.1)**	**354**	**R**	**1.2**	**(5.8,354.9)**	**360.8**	**R**	**31.7**

Table 6 shows the comparison results between MST-SPT-ERL with DR-ERL for $n = 20, 50,100,300,500,1000,2000,5000$ for the two problems.

Table 6: Comparison results between MST-SPT-ERL with DR-ERL for $n = 20, 50,100,300,500,1000,2000,5000$ for the two problems.

n	MST-SPT-ERL			DR-ERL		
	F	$F1$	T/S	F	$F1$	T/S
20	(2.8,90.6)	93.4	R	(6,92.4)	98.4	R
50	(2,235.4)	237.4	R	(5.6,237.2)	242.8	R
100	(0.8,505.6)	506.4	R	(6.8,511.2)	518	R
300	(1,1620.4)	1621.4	2.2	(7.8,1627.2)	1635	R
500	(0.2,2680.6)	2680.8	14.5	(8.4,2688)	2696.4	2.4
1000	(0.2,5435.6)	5435.8	15.5	(8.8,5444.2)	5453	9.7
2000	(0,10969.8)	10969.8	122.8	(9,10978.8)	10987.8	46.1
5000	(0,27395.8)	27395.8	1831.6	(9,27404.8)	27413.8	429.1
Av	**(0.8,6116.7)**	**6117.6**	**256.5**	**(7.6,6122.9)**	**6130.6**	**60.9**

10. Analysis and Discussions Process for BCMSP-ER AND BOMSP-EPR

1- From applying the BAB method, we notice that the EDD rule is more efficient in rewarding of efficient solution (optimal) and better CPU-time than MST rule although MST rule is useful for the function E_{max} (see tables (2) and (3)).
2- The proposed approximate method proved their efficiency by obtaining good results for the two problems (see tables (4) and (5)).
3- From table (6), we see that the heuristic method MST-SPTERL is better than the heuristic DR-ERL.

11. Conclusions and Recommendations

1- From the above tables, for 99% of the results, we see that the number of efficient solutions is (1), except for the state when $E_{max}(\text{MST}) < E_{max}(\text{EDD})$, $R_L(\text{MST}) > R_L(\text{EDD})$ and $L_{max}(\text{MST}) > L_{max}(\text{EDD})$, it may give two efficient solutions.
2- Through the previous aspect, since the efficient solution is almost unique, it represents the optimal solution to the BOMSP-EPR, so the proposed exact and approximate solution methods are applied to the BCMSP-ER only.
3- From relation (2), we have a linear relation between R_L and E_{max} and $R_L \geq E_{max}$ (unless $L_{max} < 0$) this means when L_{max} is fixed, any increasing (decreasing) in E_{max} means increasing (decreasing) in R_L and vice versa.

4- As a special case, If $p_j = d_j = p, \forall j$, then d_j has no effect on the efficient (optimal) solution of BCMSP-ER (BOMSP-EPR).
5- We suggest using the local search methods to solve the two BCMSP-ER and BOMSP-EPR.
6- We propose to add the release date (r_j) or/and setup time (S_f) constraint to our problems to obtain new problems and suggesting new solving methods, like: $1/r_j/(E_{max}, R_L)$ and $1/S_f/(E_{max}, R_L)$.
7- As future work, we suggesting studying new MSP like: $1//(T_{max}, R_L)$, $1//(E_{max}, T_{max})$ and $1//(E_{max}, T_{max}, R_L)$.

ACKNOWLEDGMENTS

The authors would like to thank Mustansiriyah University at (www.uomustansiriyah.edu.iq) Baghdad – Iraq for its support in the present work.

References

[1] F. M. E. P. D. Tapan S., "A Branch -and-Bound Approach to the Bicriterion Scheduling Problem Involving Total Flowtime and Range of Lateness," *Management Science,* vol. 34, no. 2, pp. 254-260, 1988.

[2] L. C. a. H. R., "An Algorithm for Minimizing the Range of Lateness on a Single Machine," *J. Opl Res. Soc.,* vol. 42, no. 2, pp. 183-186, 1991.

[3] B. A. M. a. K. S. Toktas, "Two Machine Flow Shop Scheduling with Two Criteria: Maximum Earliness and Makespan," *European Journal of operation Research,* vol. 157, pp. 286-295, 2004.

[4] A. M. Delphi, *Algorithms to Solve Multicriteria Scheduling problems on single Machine,* Baghdad, Iraq: M. Sc. Thesis, Mustansiriyah University, College of science, Dept. of Mathematics, 2011.

[5] I. S. J., *Single Machine Multicriteria Scheduling,* Baghdad, Iraq.: M. Sc. Thesis, Mustansiriyah University, College of science, Dept. of Mathematics, 2014.

[6] A. Z. M., *Single Machine Scheduling Problem by Using Exact and Local Search Methods,* Baghdad, Iraq: M. Sc. Thesis, Mathematics department, College of Science, Mustansiriyah University, 2015.

[7] K. R. a. K. J. J. Baker, "Job shop scheduling with modified due dates," *Oper, Manage,* vol. 4, no. 1, pp. 11-22, 1983.

[8] S. A. Ali, *A comparison of Local Search Algorithms for Multicriteria Scheduling Problems,* Baghdad, Iraq.: M. Sc. Mustansiriyah University, College of Sceince, Mathematics Dept. , 2016.

[9] A. M. Gh, *Exact and Approximation Methods for Solving Combinatorial Optimization Problems,* Baghdad, Iraq.: Ph. D. Thesis, Mathematics department, College of Science, Mustansiriyah University, 2021.

[10] a. M. G. A. Faez H. Ali, "Optimal and Near Optimal Solutions for Multi Objective Function on a Single Machine," in *1st International Conference on Computer Science and Software Engineering (CSASE2020), Sponsored by IEEE Iraq Section, 16-17 Apr. ,* Duhok, Kurdistan Region – Iraq,, 2020.

[11] F. H. A. a. M. G. Ahmed, "Local Search Methods for Solving Total Completion Times, Range of Lateness and Maximum Tardiness Problem," in *6th international engineering conference, Sustainable Technology and Development, IEEE, (IEC-2020),* Erbil, Iraq., 2020.

[12] J. R. Jakson, "Scheduling a Production Line to Minimize Maximum Tardiness," Res. Report 43 Management science, Res. Project, University of California, Loss Angles, CA., 1955.

[13] F. H. A. W. S. H. Aseel Aboud Jawad, "Using Heuristic and Branch and Bound Methods to Solve Multi-Criteria Machine Scheduling Problem," *Iraqi Journal of Science,* vol. 61, no. 8, pp. 2055-2069, 2020.

[14] F. H. Ali and A. A. Jawad, "Minimizing the Total Completion Time and Total Earliness Time Functions for Machine Scheduling Problem Using Local Search Methods," *Iraqi Journal of Sceince*, vol. Special Issue, pp. 126-133, 2020.