Journal of Science and Engineering Applications Vol. 8, No.1, pp. 122–131, June 2026

Journal of Science and Engineering Applications

Contents are available at https://jsea.iujournals.com

Theoretical Model for the Uniform Machines Scheduling with release dates Under Multi-Objective Optimization

Marwa B. kadhem [1], Sajjad majeed jasim [1], Hanan Ali chachan [1*]

[1]Department of Mathematics, College of Science, University of Mustansiriyah., Baghdad, Iraq.

ARTICLE INFORMATION

Received date: 29-04-2026
Revised date: 24-05-2026
Accepted date: 10-06-2026

Keywords

Uniform parallel machine
Release date
Multi Objective Function
Dominance properties

ABSTRACT

The scheduling of parallel machines has attracted significant attention due to its wide utilization in the fields of manufacturing and data processing. This article presents an in-depth analysis of Uniform Parallel Machine Scheduling (UPMS) problem, considering sequence-dependent job characteristics with release dates are imposed. It focuses on efficiently scheduling n jobs across m machines to achieve minimal overall performance penalties that integrates total flow time, tardiness, earliness, and late work. From a practical's and point it is important since most real manufacturing settings involve multiple machines within each workstation, however, efficient solution methods are still scarce, and the optimal scheduling algorithms currently available for multiprocessor systems are limited in their ability to address large scale problems. As a result, practitioners tend to adopt methods based on dominance properties to guide solution selection. This leads to key questions: Are current schedules sufficiently efficient? Do dominance properties effectively support optimal decision-making in multi-objective UPMS problems? How close are the solutions to optimality?

This paper focuses on the NP-hard problem of scheduling n jobs on uniform parallel machines by proposing a dominance-based approach. We review the existing results under specific conditions and introduce several new solvable cases. In addition, we identify key research gaps and suggest direction for future work. The proposed method is evaluated using a comprehensive set of randomly generated test instances, demonstrating its effectiveness across a wide range of problem scenarios.

1. Introduction

Parallel machine scheduling problem (PMSP), where jobs tend to be executed by machines, the usage of smart association becomes an base demand, motives of coordination arises an assortment of operations will be remedy by

assortment of parallel machines , with a range of objectives reach good feasible solutions in a short computational time, providing with a decision support production problems that emerged in practical cases from the in an assortment of industrial settings, including auto plants, the mechanical, pharmaceutical, chemical and all science filed by Fanjul-Peyroand Ruiz(2010) [1] and Larroche et al (2022) [2].

The beginning relevant explore by McNaughton (1959) [3], various PMSPs have attracted extend look after among researchers.

A ranking of a scheme for parallel machine scheduling problem come by Graham et al (1979) [4] and Edis (2009) [5]. this one scheme could utilize a three field event style $\boldsymbol{\alpha} \mid \boldsymbol{\beta} \mid \boldsymbol{\gamma}$. Specific features of the machine environment ($\boldsymbol{\alpha}$), job characteristics and constraints ($\boldsymbol{\beta}$) and global optimality criterion ($\boldsymbol{\gamma}$). the system style environments classification, there are three parallel machine environments: as identical, uniform, unrelated basically depends on the features of parallel machines by Cheng and Sin (1990)[6].

Owing the challenges of scheduling uniform parallel machines (UPMS) plays a crucial the production operation represent , This has prompted the attention of many examine carefully into finding an optimal scheduling in recent decades, the target of minimizing the weighted flow time. A branch and bound algorithm are produced and also a great deal of effective mathematical solutions for uniform parallel machines scheduling that have scheduling multi criteria, I. Dessouky et al (1990). and M. Dessouky(1998) works [7], [8]. .A branch-and-bound technique was applied to solve these problems that hand out by Azizoglu and Kirca(1999) [9], the objective Liao and Lin (2003) [10] is binding To decrease the makespan for two machine for scheduling uniform parallel challenges (UPMS).the UPMSP was developed in to a unique identical PMSP a strategy for handing dissolve it in the most suitable method. The Mallek and Boudhar(2024) suggested reducing the makespan utilized for scheduling on contention graph uniform parallel machines by designing a mixed-integer line program (MILP) with lower and upper bounds [11]. Song et al.(2024) [12] be decided a polynomial-time strategy for preemptive scheduling at uniform machine and a two-approximation strategy for non-preemptive on the aforesaid machine for a green manufacturing system have mentioned.
A single performance measuring criterion was the focus of widely the several achievement were granted in the untimely years of the theory of scheduling's discovery. However, due to scheduling theory's growing influence on decision-making across a variety of real-world contexts, including industrial production and services, which frequently call for many criteria to assess performance quality, [13], [14], [15], [16], [17],[18].

This constraint submits the issue of schedule (n) jobs in machines scheduling on the uniform problem withrelease dates and varied fulfillment measures withThe design of lowering the sum of total flow time, total tardiness, total earliness, and total late work this problem is represented by $\mathbf{Q_m} / \mathbf{r_i} / \sum_{\mathbf{i=1}}^{\mathbf{m}} \sum_{\mathbf{j=1}}^{\mathbf{n}} (\mathbf{C_{ij}} + \mathbf{T_{ij}} + \mathbf{E_{ij}} + \mathbf{v_{ij}})$
Before confer about our challenge, we feel it is substantial to existent with respect to research from the written works that included parallel machines scheduling problems either one or increased of the execution criteria included in our trouble. This essay we execute the scheduling problem with four criteria by measuring execution accompanied by the release time. To the finest of our potential, this one matter has not been designed.
his paper is structured as follows: Description of the Problem in Section 2. Mathematical Programming with Integer Constraintsis described in Section 3. We provide dominance properties related to the problem in Section 4. Finally, Section 5 discusses the presents our conclusions and some suggestions additional research.

2. Description of the Problem

Here, we introduce the notation system employed for characterization and subsequent to solve for Q_m /r_j /$\sum_{i=1}^{m} \sum_{j=1}^{n} (C_{ij} + T_{ij} + E_{ij} + v_{ij})$ problem. we adopt the following notations :

j A Series of tasks.

i A Series of machines.

n A Series of tasks (n = |j|).

m A Series of the machines (m = |M|).

s_i Relative processing speed of machine i

p_{ij} Processing time of task j on machine i where p_{ij}/s_i for uniform parallel machine.

r_j Release date of job j

d_j Due date of job j.

σ_i Partial schedule sequence of the jobs assigned to machine i.

C_{ij} (σ) Completion time of job j in partial schedule σ_i, $c_{i1} = r_1 + p_{ij}$, $c_{ij} = \max\{ r_j, c_{ij-1}\} + p_{ij}$, j =2,…,n

T_{ij} (σ) Tardiness of job j in partial schedule σ_i, $T_{ij} = \max\{ c_{ij} - d_j, 0\}$.

E_{ij} (σ) Earliness of job j in partial schedule σ_i, E_{ij} = max $\{d_j - c_{ij}, 0\}$.

v_{ij} (σ) late work of job j in partial scheduleσ_i, v_{ij} = min $\{T_{ij},\ p_{ij}\}$.

3. Mathematical Programming with Integer Constraints

The $Q_m/r_j/\sum_{i=1}^{m}\sum_{j=1}^{n}(C_{ij}+T_{ij}+E_{ij}+v_{ij})$ problem may be modelled as a hypothetical integer programming problem (P).

$$Z_p = min\sum_{k=1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{n}(C_{jk}^{t}X_{jk}^{t} + T_{jk}^{t}X_{jk}^{t} + E_{jk}^{t}X_{jk}^{t} + v_{jk}^{t}X_{jk}^{t} \quad (3.1)$$

Subject to

$$\sum_{k=1}^{m}\sum_{t=1}^{n}X_{jk}^{t} = \qquad j = 1,2,\dots,n\} \qquad (3.2)$$

$$\sum_{j=1}^{n}X_{jk}^{t} \leq 1 \qquad \left.\begin{matrix} t = 1,2,\dots,n \\ k = 1,2,\dots,m \end{matrix}\right\} \qquad (3.3)$$

$$C_{1k}^{t} = (r_1 + p_{lk}^{t})$$

$$C_{jk}^{t} = max(r_j, C_{j-1,k}^{t}) \qquad \left.\begin{matrix} j = 2,3,\dots,n \\ k = 1,\dots,m \\ t = 1,\dots,n \end{matrix}\right\} \qquad (3.4)$$

$$T_{jk}^{t} = \max(0, C_{jk}^{t} - d_j) \qquad \left.\begin{matrix} j,t = 1,2,\dots,n \\ k = 1,2,\dots,m \end{matrix}\right\} \qquad (3.5)$$

$$E_{jk}^{t} = \max(0, d_j - C_{jk}^{t}) \qquad \left.\begin{matrix} j,t = 1,2,\dots,n \\ k = 1,2,\dots,m \end{matrix}\right\} \qquad (3.6)$$

$$v_{jk}^{t} = \min\{T_{jk}^{t}, p_{lk}^{t}\} \qquad \left.\begin{matrix} j,t = 1,2,\dots,n \\ k = 1,2,\dots,m \end{matrix}\right\} \qquad (3.7)$$

$$X_{jk}^{t} = 0 \text{ or } 1 \qquad \left.\begin{matrix} j,t = 1,2,\dots,n \\ k = 1,2,\dots,m \end{matrix}\right\} \qquad (3.8)$$

Where C_{jk}^{t} confirmed the completion time of job j is scheduling on machine k at t^{th} position and

$$X_{jk}^{t} = \begin{cases} 1 \; if\ job\ j\ scheduled\ in\ t^{th} position\ on\ machine\ k \\ 0 \; o.w \end{cases}$$

With respect to Constraint (3.2)each state occupies at most one placement on the machine, by contrast constraint (3.3) the model enforces the allocation at most of one job per position on each machine. Specifies the completion time of job j when processed in position t on machine k Constraint (3.4). Constraint (3.5), (3.6) and($\mathbf{3.7}$) grant tardiness, earliness and late work of job j, whenever it is completion time $\boldsymbol{C_{jk}^{t}}$in the $\boldsymbol{th}$ placement on the machine k .To conclude, constraint (3.8) is a variable is intended to show job j assigned on machine k in placement t, then $\boldsymbol{X_{jk}^{t} = 1}$ otherwise 0.

4. Dominance Rules

We establish a number of dominance rules for Q_m $/r_j/\sum_{i=1}^{m}\sum_{j=1}^{n}(C_{ij} + T_{ij} + E_{ij} + v_{ij})$ the problem has been established in theorem (4.1).

Let S_{1k}=(β_1 jk β_2) and S_{2k}=($\beta_1 kj$ β_2) think about two sequences :β_1 and β_2, that infrequent configuration considers the remaining n-2 ,and the two jobs (j) and (k) which are adjacent jobs using same machine with $p_{ij} \leq p_{ik}$ and $d_j \leq d_k$.

Let Φ represent completion time of β_1 and c which coincide with the start time of the first job in set β_2 starts, Let δ_{ijk} be value of functions on which δ_{ijk}=$\sum_{i=1}^{m}\sum_{j=1}^{n}(C_{ij} + T_{ij} + E_{ij} + v_{ij})$ for both jobs, subsequences (j), and (k) when (j) precedes (k) and δ_{ijk} are functions value for the subsequences of both jobs (j) and (k) when (k) comes before (j).

Right now, we proceed to analyse the adjustments in Δ_{ijk}=$\delta_{ijk} - \delta_{ikj}$ and, with this subsequent instances $\Delta_{ijk} \leq 0$ it displays that

- With respect to $\Delta_{ijk} < 0$ subsequently, at timeΦ, job j needs to be before job k.
- With respect to $\Delta_{ijk} > 0$ subsequently, at timeΦ, job k needs to be before job j.
- With respect to $\Delta_{ijk} = 0$ subsequently, there is no little change in schedule (j) or (k) first.

Theorem(4.1): About the Q_m $/\boldsymbol{r_j}/\sum_{\boldsymbol{i=1}}^{\boldsymbol{m}}\sum_{\boldsymbol{j=1}}^{\boldsymbol{n}}(\boldsymbol{C_{ij}} + \boldsymbol{T_{ij}} + \boldsymbol{E_{ij}} + \boldsymbol{v_{ij}})$ problem if $\boldsymbol{\mathcal{P}_{ij} \leq \mathcal{P}_{ik}}$ for all i=1,2,…,m, , and $\boldsymbol{d_j \leq d_k}$where the jobs ($j$) and ($k$) are two adjacent jobs on the same machine ,then the job (j) comes before the job (k) in optimal sequences.

First : this situation are classified categorized in to the following categories if $r \leq \Phi$ is true

Case 1: If $d_j \leq \Phi + \mathcal{P}_{ij}, d_k \leq \Phi + \mathcal{P}_{ik}$, jobs ($j$) and ($k$) at all times tardy.

Proof: Thus, we proved this one S_{1k} domains S_{2k}**,** All we have to complete are display Δ_{ijk}=$\delta_{ijk} - \delta_{ikj}$ both jobs (j) and (k) are always tardy then E_{ij}= $E_{ik} =$ 0 and $v_{ij} = \mathcal{P}_{ij}, v_{ik} = \mathcal{P}_{ik}$.

δ_{ijk}=$[(\Phi+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-d_k)+0+\mathcal{P}_{ik})]$

$$=[4\Phi+5\mathcal{P}_{ij}+3\mathcal{P}_{ik}-d_j-d_k-2\mathfrak{r}]$$

$$\delta_{ikj}=[(\Phi+\mathcal{P}_{ik})-\mathfrak{r}+(\Phi+\mathcal{P}_{ik})-d_k)+0+\mathcal{P}_{ik}+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij})-d_j)+0+\mathcal{P}_{ij}]$$

$$=[4\Phi+5\mathcal{P}_{ik}+3\mathcal{P}_{ij}-d_j-d_k-2\mathfrak{r}]$$

$$\Delta_{ijk}=\delta_{ijk}-\delta_{ikj}$$

$$=-2\mathcal{P}_{ik}+2\mathcal{P}_{ij}\le 0$$

Since $\mathcal{P}_{ij}<\mathcal{P}_{ik}$, then $j \rightarrow k$

Case 2: If $d_j\le\Phi+\mathcal{P}_{ij}$, $\Phi+\mathcal{P}_{ik}\le d_k\le\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik}$, job (j) be constantly tardy and job (k) is tardy assuming that not scheduled at first.

Proof: since job j is tardy thus $E_{ij}=0$, job (k)is tardy if not scheduled first $T_{ik}=0$ and $\in V_{ik}\{c_{ik}-d_{ik},0\}$ (i.e. job (k) is early or partial early)

(a) When $V_{ik}=0$

$$\Delta_{ijk}=[(\Phi+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-d_k)+o+\mathcal{P}_{ik})]$$

$$=[4\Phi+3\mathcal{P}_{ik}+5\mathcal{P}_{ij}-d_j-d_k-2\mathfrak{r}]$$

$$\delta_{ikj}=[(\Phi+\mathcal{P}_{ik})-\mathfrak{r}+0+(d_k-\Phi-\mathcal{P}_{ik})+0+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}]$$

$$=[2\Phi+3\mathcal{P}_{ij}+2\mathcal{P}_{ik}-d_j+d_k-2\mathfrak{r}]$$

$$\Delta_{jki}=\delta_{jki}-\delta_{kji}$$

$$=2\Phi+2\mathcal{P}_{ij}+\mathcal{P}_{ik}-2d_k>0$$

then job k preced job j.

(b) When $V_{ik}=c_{ik}-d_{ik}=\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik}-d_{ik}$

$$\delta_{ijk}=[(\Phi+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}-d_{ij})+0+\mathcal{P}_{ij}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-\mathfrak{r}+((\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-d_{ik})+0+\mathcal{P}_{ik})]$$

$$=[4\Phi+3\mathcal{P}_{ik}+5\mathcal{P}_{ij}-d_{ij}-d_{ik}-2\mathfrak{r}]$$

$$\delta_{ikj}=[(\Phi+\mathcal{P}_{ik})-\mathfrak{r}+0+(d_{ik}-\Phi-\mathcal{P}_{ik})+(\Phi+\mathcal{P}_{ik}-d_{ik})+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij}-d_{ij})+0+\mathcal{P}_{ij}]$$

$$=[3\Phi+3\mathcal{P}_{ij}+3\mathcal{P}_{ik}-d_{ij}-2\mathfrak{r}]$$

$$\Delta_{ijk}=\delta_{ijk}-\delta_{ikj}$$

$$=\Phi+2\mathcal{P}_{ij}-d_{ik}>0$$

then job k preced job j.

Case 3: If $d_j\le\Phi+\mathcal{P}_{ij}$, $\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik}\le d_k$, then(j) be constantly tardy and the (k) be constantly early.

Proof: Since(j) be constantly tardy and(k) be constantly early then $E_{ij}=0$,

$$T_{ik}=0 \text{ and } V_{ik}\in\{c_{ij}-d_j,0\}$$

(a) When $V_{ik}=0$ (i.e k is early)

$$\delta_{ijk}=[(\Phi+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-\mathfrak{r}+0+(d_k-\Phi-\mathcal{P}_{ij}-\mathcal{P}_{ik})+0]$$

$$=[2\Phi+3\mathcal{P}_{ij}-d_j+d_k-2\mathfrak{r}]$$

$$\delta_{ikj}=[(\Phi+\mathcal{P}_{ik})-\mathfrak{r}+0+(d_k-\Phi-\mathcal{P}_{ik})+0+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ik}+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}]$$

$$=[2\Phi+3\mathcal{P}_{ij}+2\mathcal{P}_{ik}-d_j+d_k-2\mathfrak{r}]$$

$$\Delta_{ijk}=\delta_{ijk}-\delta_{ikj}$$

$$=-2\mathcal{P}_{ik}\le 0,$$

Since job j preced job k.

(b) When $V_{ik}=c_{ik}-d_k=\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik}-d_k$

$$\delta_{ijk}=[(\Phi+\mathcal{P}_{ij})-\mathfrak{r}+(\Phi+\mathcal{P}_{ij}-d_j)+0+\mathcal{P}_{ij}+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik})-\mathfrak{r}+0+(d_k-\Phi-\mathcal{P}_{ij}-\mathcal{P}_{ik})+(\Phi+\mathcal{P}_{ij}+\mathcal{P}_{ik}-d_k)]$$

$$=[3\Phi+4\mathcal{P}_{ij}+\mathcal{P}_{ik}-d_j-2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d_k) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_j - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$= \mathcal{P}_{ij} - 2\mathcal{P}_{ik} \leq 0 \rightarrow \mathcal{P}_{ij} < \mathcal{P}_{ik}$$

Since job j precede job k.

Case 4: If $\Phi + \mathcal{P}_{ij} \leq d_j \leq d_k \leq \Phi + \mathcal{P}_{ik}$, (i.e. the job (k) be constantly tardy and (j) is tardy assuming that not scheduled firstly.

Proof: Since $E_{ik} = 0$ if job j schedule first

Then $T_{ij} = 0$ and $V_{ij} \in \{c_{ij} - d_j, 0\}$

(a) When $V_{ij} = 0$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [2\Phi + 3\mathcal{P}_{ik} + 2\mathcal{P}_{ij} + d_j - d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [(\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (\Phi + \mathcal{P}_{ik} - d_k) + \mathcal{P}_{ik} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [4\Phi + 3\mathcal{P}_{ij} + 5\mathcal{P}_{ik} - d_j - d_k - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$= (-2\Phi - \mathcal{P}_{ij} - 2\mathcal{P}_{ik} - 2d_j)$$

Since job j precede job k

(b) When $V_{ij} = c_{ij} - d_k$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d_j) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [3\Phi + 3\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [(\Phi + \mathcal{P}_{ik}) - r + (\Phi + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - r + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [4\Phi + 3\mathcal{P}_{ij} + 5\mathcal{P}_{ik} - d_j - d_k - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$= -\Phi - 2\mathcal{P}_{ik} + d_j \leq 0$$

, then job j preced job k.

Case 5: If $\Phi + \mathcal{P}_{ij} \leq d_j$, $\Phi + \mathcal{P}_{ik} \leq d_k \leq \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}$, (i.e. both of the two jobs (j) and (k) are tardy assuming that they not scheduled first).

Proof:

(a) When $V_{ij} = 0, V_{ik} = 0$,

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [2\Phi + 2\mathcal{P}_{ij} + 3\mathcal{P}_{ik} + d_j - d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [2\Phi + 3\mathcal{P}_{ij} + 2\mathcal{P}_{ik} - d_j + d_k - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$= \mathcal{P}_{ik} - \mathcal{P}_{ij} + 2d_j - 2d_k$$

Since $\Phi + \mathcal{P}_{ij} \leq d_j$ and $\Phi + \mathcal{P}_{ik} \leq d_k$

$$\rightarrow \mathcal{P}_{ik} - \mathcal{P}_{ij} + 2d_j - 2d_k \leq 0$$

then job k preced job j.

(b) When $V_{ij} = c_{ij} - d_j, V_{ik} = c_{ik} - d_k$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d_j) + (\Phi + \mathcal{P}_{ij} + p_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_k - 2\mathfrak{r}]$$

$$\delta_{kji} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d_k) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_j + 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$= -d_j - d_k \le 0,$ then job k preced job j

(c) When $V_{ij} = 0, V_{ik} = c_{ik} - d_k$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [2\Phi + 2\mathcal{P}_{ij} + 3\mathcal{P}_{ik} + d_j - d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d_k) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_j - 2\mathfrak{r}]$$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= -\Phi - \mathcal{P}_{ij} + 2d_j - d_k \le 0$, then job k preced job j.

(d) When $V_{ij} = c_{ij} - d_{ij}, V_{ik} = 0$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d_j) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k) + 0 + \mathcal{P}_{ik}]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [2\Phi + 3\mathcal{P}_{ij} + 2\mathcal{P}_{ik} - d_j + d_k - 2\mathfrak{r}]$$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= \Phi + \mathcal{P}_{ik} + d_j - 2d_k$, Since $\Phi + \mathcal{P}_{ik} \le d_k$

$= \Phi + \mathcal{P}_{ik} + d_j - 2d_k \le 0,$ then jobj preced job k.

Case 6: If $\Phi + \mathcal{P}_{ij} \le d_j \le \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} \le d_k$ (i.e. job (j) is tardy assuming that not scheduled first, the job (k) be constantly early)

Proof:

(a) when $V_{ij} = 0, V_{ik} = 0$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + 0]$$

$$= [d_j + d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [2\Phi + 3p_{ij} + 2\mathcal{P}_{ik} - d_j + d_k - 2\mathfrak{r}]$$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= -2\Phi - 3\mathcal{P}_{ij} - 2\mathcal{P}_{ik} + 2d_j \le 0,$

then jobj preced job k.

(**b**)when $V_{ij} = \Phi + \mathcal{P}_{ij} - d_j,$

$$V_{ik} = \begin{cases} \Phi + \mathcal{P}_{ik} - d_k & \text{if (k)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k & \text{if (k)is } 2^{nd} \end{cases}$$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d_j) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k)]$$

$$= [2\Phi + 2\mathcal{P}_{ij} + \mathcal{P}_{ik} - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d_k) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j) + 0 + \mathcal{P}_{ij})]$$

$$= [3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d_j - 2\mathfrak{r}]$$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= -\Phi - \mathcal{P}_{ij} - 2\mathcal{P}_{ik} + d_j \le 0,$

then jobj preced job k.

Case 7: If $\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} \le d_j \le d_k$,(i.e. both (j) and (k) are early)

Proof:

(a) when $V_{ij} = 0, V_{ik} = 0$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + 0]$$

$$= [d_j + d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + 0 + \left(\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}\right) - \mathfrak{r} + 0 + \left(d_j - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}\right) + 0 \,)$$

$$= [d_j + d_k - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$\Delta_{ijk} = [d_j + d_k - 2\mathfrak{r}] - [d_j + d_k - 2\mathfrak{r}] = 0,$$

thenjob j or k first is irrelevant

(**b**) when $V_{ij} = \begin{cases} \Phi + \mathcal{P}_{ij} - d_j & \text{if (j)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_j & \text{if (j)is } 2^{nd} \end{cases}$,

$$V_{ik} = \begin{cases} \Phi + \mathcal{P}_{ik} - d_k & \text{if (k)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k & \text{if (k)is } 2^{nd} \end{cases}$$

$$\delta_{ijk} = [\left(\Phi + \mathcal{P}_{ij}\right) - \mathfrak{r} + 0 + \left(d_j - \Phi - \mathcal{P}_{ij}\right) + \left(\Phi + \mathcal{P}_{ij} - d_j\right) + \mathfrak{r} + 0 + \left(d_k - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}\right) + \left(\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_k\right)]$$

$$= [2\Phi + 2\mathcal{P}_{ij} + \mathcal{P}_{ik} - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d_k) + \left(\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}\right) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ik} - \mathcal{P}_{ij}) + \left(\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_j\right)]$$

$$= [\,2\Phi + \mathcal{P}_{ij} + 2\mathcal{P}_{ik} - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$= \mathcal{P}_{ij} - \mathcal{P}_{ik} \leq 0$,then job j comes befor j.

(**c**) when $V_{ij} = \begin{cases} \Phi + \mathcal{P}_{ij} - d_j & \text{if (j)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d_j & \text{if (j)is } 2^{nd} \end{cases}$,

$$V_{ik} = 0$$

$$\delta_{ijk} = [\left(\Phi + \mathcal{P}_{ij}\right) - \mathfrak{r} + 0 + \left(d_j - \Phi - \mathcal{P}_{ij}\right) + \left(\Phi + \mathcal{P}_{ij} - d_j\right) + \left(\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}\right) - \mathfrak{r} + 0 + \left(d_k - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}\right) + 0]$$

$$= [\Phi + \mathcal{P}_{ij} + d_k - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d_k - \Phi - \mathcal{P}_{ik}) + 0 + \left(\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}\right) - \mathfrak{r} + 0 + \left(d_j - \Phi - \mathcal{P}_{ik} - \mathcal{P}_{ij}\right) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d_j)]$$

$$= [\,\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} + d_k - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$= -\mathcal{P}_{ik} \leq 0$, then job j comes befor j.

Second : Assuming that $\mathrm{r} > \Phi$,then we demonstrato that the theorem is correct within same technique as we did earlier.

Theorem (4.2): The jobs(j), besides (k) are adjacent jobs in the trouble $\mathcal{P}$ if $\mathfrak{r}_j = \mathfrak{r}$ and $d_j = d$ for whole ($i \in n$) also if $p_{ij} \leq p_{ik}$ besides $d_j = d$, then then job (j) should precedes job (k) for at least one sequences with the optimum value for trouble p

Proof: By resing Same regulation as previousty we can demonstrate the viability of that assumption in the following envelope:

First $\mathfrak{r} \leq \Phi$

Case 1: If $d \leq \Phi + \mathcal{P}_{ij} \leq \Phi + \mathcal{P}_{ik}$ **(**i.e. both of the jobs (j), besides (k) are repeatedly tardy)

Proof: Since the jobs (j), besides (k) are both tardy then $E_{ji} = E_{ki} = 0$ and $v_{ij} = \mathcal{P}_{ij}$, $v_{ik} = \mathcal{P}_{ik}$

$$\delta_{ijk} = [\left(\Phi + \mathcal{P}_{ij}\right) - \mathfrak{r} + \left(\Phi + \mathcal{P}_{ij} - d\right) + 0 + \mathcal{P}_{ij} + \left(\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}\right) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d) + 0 + \mathcal{P}_{ik}]$$

$$= [4\Phi + 5\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - 2d - 2\mathfrak{r}]$$

$$\delta_{ikj} = [(\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} - d) + 0 + \mathcal{P}_{ik} + \left(\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}\right) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij})]$$

$$= [\,4\Phi + 5\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - 2d - \mathfrak{r})]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$= 2\mathcal{P}_{ij} - 2\mathcal{P}_{ik} \leq 0$, then job j comes before k.

Case 2: If $d \leq \Phi + \mathcal{P}_{ij}, \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} \leq d$ **,** the job (j) is tardy and the job (k) is early repeatedly)

Proof: As the job (j) is tardy repeatedly, then $E_{ij} = 0$, besides the job (k) is alwayes on time, then $T_{ik} = 0, V_{ik} = \{c_{ik} - d, 0\}$

(a) When $V_{ik} = 0$

$$\delta_{ijk} = [\left(\Phi + \mathcal{P}_{ij}\right) - \mathfrak{r} + \left(\Phi + \mathcal{P}_{ij} - d\right) + 0 + \mathcal{P}_{ij} + \left(\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}\right) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + 0]$$

$= \left[2\Phi + 3\mathcal{P}_{ij} - 2\mathfrak{r}\right]$

$\delta_{ikj} = [(\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij}]$

$= \left[2\Phi + 2\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - 2\mathfrak{r}\right]$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= -2\mathcal{P}_{ik}$ then job j precede job k.

(b) When $V_{ik} = c_{ik} - d$

$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij} + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d)]$

$= \left[3\Phi + 4\mathcal{P}_{ij} + \mathcal{P}_{ik} - d - 2\mathfrak{r}\right]$

$\delta_{ikj} = [(\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij}]$

$= \left[3\Phi + 3\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - d - 2\mathfrak{r}\right]$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= \mathcal{P}_{ij} - 2\mathcal{P}_{ik} \leq 0$, then job j precede job k.

Case 3: If $\Phi + \mathcal{P}_{ij} \leq \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} \leq d$(i.e. job (j) is tardy if not scheduled first besides job(k)is repeatedly early)

Proof:

(a) when $V_{ij} = 0, V_{ik} = 0$

$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + 0]$

$= [2d - 2\mathfrak{r}]$

$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij})]$

$= \left[2\Phi + 2\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - 2\mathfrak{r}\right]$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= [2d - 2\mathfrak{r}] - \left[2\Phi + 2\mathcal{P}_{ik} + 3\mathcal{P}_{ij} - 2\mathfrak{r}\right] = 0$

$= -2\Phi - 2\mathcal{P}_{ik} - 3\mathcal{P}_{ij} + 2d \leq 0$

(**b**)when $V_{ij} = \Phi + \mathcal{P}_{ij} - d$,

$$V_{ik} = \begin{cases} \Phi + \mathcal{P}_{ik} - d & \text{if (k)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d & \text{if (k)is } 2^{nd} \end{cases}$$

$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d)$

$= \left[2\Phi + 2\mathcal{P}_{ij} + \mathcal{P}_{ik} - 2\mathfrak{r}\right] =$

$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij} - d) + 0 + \mathcal{P}_{ij})]$

$= \left[3\Phi + 3\mathcal{P}_{ij} + 3\mathcal{P}_{ik} - d - 2\mathfrak{r}\right]$

$\Delta_{jki} = \delta_{jki} - \delta_{kji}$

$= -\Phi - \mathcal{P}_{ij} + 2\mathcal{P}_{ik} + d \leq 0$

Case 4: If $\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ki} \leq d$ (both jobs (j)and (k) are early)

Proof:

(a) when $V_{ij} = 0, V_{ik} = 0$

$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij}) + 0 + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik} + 0)]$

$= [2d - 2\mathfrak{r}]$

$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + 0 + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik} - \mathcal{P}_{ij}) + 0]$

$= [2d - 2\mathfrak{r}]$

$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$

$= [2d - 2\mathfrak{r}] - [2d - 2\mathfrak{r}] = 0$

then scheduling jor k is irrelevant

$$(\mathbf{b})\text{when } V_{ij} = \begin{cases} \Phi + \mathcal{P}_{ij} - d & \text{if (j)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d & \text{if (k)is } 2^{nd}, \end{cases}$$

$$V_{ik} = \begin{cases} \Phi + \mathcal{P}_{ik} - d & \text{if (k)is } 1^{st} \\ \Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d & \text{if (k)is } 2^{nd} \end{cases}$$

$$\delta_{ijk} = [(\Phi + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d_j - \Phi - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} - d) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ij} - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d)$$

$$= [2\Phi + 2\mathcal{P}_{ij} + \mathcal{P}_{ik} - 2\mathfrak{r}]$$

$$\delta_{ikj} = [((\Phi + \mathcal{P}_{ik}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik}) + (\Phi + \mathcal{P}_{ik} - d) + (\Phi + \mathcal{P}_{ik} + \mathcal{P}_{ij}) - \mathfrak{r} + 0 + (d - \Phi - \mathcal{P}_{ik} - \mathcal{P}_{ij}) + (\Phi + \mathcal{P}_{ij} + \mathcal{P}_{ik} - d)]$$

$$= [2\Phi + \mathcal{P}_{ij} + 2\mathcal{P}_{ik} - 2\mathfrak{r}]$$

$$\Delta_{ijk} = \delta_{ijk} - \delta_{ikj}$$

$$= \mathcal{P}_{ij} - \mathcal{P}_{ik} \leq 0, \text{then job j come before job k}$$

5. Conclusion

Our study essential the scheduling of independent jobs on uniform parallel computers with the attempt to minimize multi-objective function criteria. To define the optimal schedules, we created dominance rules. Many areas need to be focused on further, some of which are extensions of the findings in a broader sense: Heuristic processes for more complex issues, homogenous and unrelated parallel machine methods, for multi-objective functions and the want to find tighter lower bounds and more effective dominance rules that provide excellent results in a manageable amount of CPU time. Furthermore, numerous limitations or characteristics of scheduling problems in real systems must be considered in order to use scheduling algorithms designed for the challenges. By way of illustration, in many pure parallel machine shops, such as the drilling workstation in electronic circuit board production systems, setup ways must be conducted before.

Future research may consider more general applications of its branching and restriction methods. Additionally, the development of upper bounds and the derivation of lower bounds should be explored in future analyses.

Conflicts Of Interest

The authors assure that they have no collision of interest.

Acknowledgment

The authors gratefully acknowledge the support of Mustansiriyah University, college of science, Department of math, in the preparation of this article .

References

[1] L. Fanjul-Peyro and R. Ruiz, "Iterated greedy local search methods for unrelated parallel machine scheduling," *Eur. J. Oper. Res.*, vol. 207, no. 1, pp. 55–69, 2010.

[2] F. Larroche, O. Bellenguez, and G. Massonnet, "Clustering-based solution approach for a capacitated lot-sizing problem on parallel machines with sequence-dependent setups," *Int. J. Prod. Res.*, vol. 60, no. 21, pp. 6573–6596, 2022.

[3] R. McNaughton, "Scheduling with deadlines and loss functions," *Manage. Sci.*, vol. 6, no. 1, pp. 1–12, 1959.

[4] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. R. Kan, "Optimization and approximation in deterministic sequencing and scheduling: a survey," in *Annals of discrete mathematics*, vol. 5, Elsevier, 1979, pp. 287–326.

[5] E. B. Edis, "Resource constrained parallel machine scheduling problems with machine eligibility restrictions: Mathematical and constraint programming based approaches," 2009, *Dokuz Eylul Universitesi (Turkey)*.

[6] T. C. E. Cheng and C. C. S. Sin, "A state-of-the-art review of parallel-machine scheduling research," *Eur. J. Oper. Res.*, vol. 47, no. 3, pp. 271–292, 1990.

[7] M. I. Dessouky, B. J. Lageweg, J. K. Lenstra, and S. L. van de Velde, "Scheduling identical jobs on uniform parallel machines," *Stat. Neerl.*, vol. 44, no. 3, pp. 115–123, 1990.

[8] M. M. Dessouky, "Scheduling identical jobs with unequal ready times on uniform parallel machines to minimize the maximum lateness," *Comput. Ind. Eng.*, vol. 34, no. 4, pp. 793–806, 1998.

[9] M. Azizoglu and O. Kirca, "On the minimization of total weighted flow time with identical and

uniform parallel machines," *Eur. J. Oper. Res.*, vol. 113, no. 1, pp. 91–100, 1999.

[10] C.-J. Liao and C.-H. Lin, "Makespan minimization for two uniform parallel machines," *Int. J. Prod. Econ.*, vol. 84, no. 2, pp. 205–213, 2003.

[11] A. Mallek and M. Boudhar, "Scheduling on uniform machines with a conflict graph: complexity and resolution," *Int. Trans. Oper. Res.*, vol. 31, no. 2, pp. 863–888, 2024.

[12] J. Song, C. Miao, and F. Kong, "Uniform-machine scheduling problems in green manufacturing system," *Math. Found. Comput.*, p. 0, 2024.

[13] A. S. Hameed, H. A. Chachan, and H. J. Ali, "Optimizing multi-objective problem with setup times," in *AIP Conference Proceedings*, AIP Publishing, 2023.

[14] F. H. Ali, H. A. Chachan, and S. B. Abdulkareem, "Neural network for minimizing tricriteria objective function for machine scheduling problem," in *Journal of Physics: Conference Series*, IOP Publishing, 2021, p. 32023.

[15] H. A. Chachan, F. H. Ali, and M. H. Ibrahim, "Branch and Bound and Heuristic Methods for Solving Multi-Objective Function for Machine Scheduling Problems," in *2020 6th International Engineering Conference "Sustainable Technology and Development"(IEC)*, IEEE, 2020, pp. 109–114.

[16] H. A. Chachan, H. M. A. R. Al-Badri, and W. M. Elaibi, "Optimality analysis of fuzzy critical path for projects (an applied study)," *Ital. J. Pure Appl. Math.*, p. 400.

[17] M. H. Ibrahim, F. H. Ali, and H. A. Chachan, "Dominance rules for single machine scheduling to minimize weighted multi objective functions," in *AIP Conference Proceedings*, AIP Publishing, 2023.

[18] J. Wang, D. Lv, J. Xu, P. Ji, and F. Li, "Bicriterion scheduling with truncated learning effects and convex controllable processing times," *Int. Trans. Oper. Res.*, vol. 28, no. 3, pp. 1573–1593, 2021.