AIMS Mathematics

AIMS Mathematics, 8(7): 17259–17271.
DOI: 10.3934/math.2023882
Received: 22 March 2023
Revised: 23 April 2023
Accepted: 23 April 2023
Published: 18 May 2023

http://www.aimspress.com/journal/Math

Research article

On employing pythagorean fuzzy processing time to minimize machine rental cost

Salwa El-Morsy[1,2], Junaid Ahmad[3] and Reny George[4,*]

[1] Department of Mathematics, College of Science and Arts, Qassim University, Al-Badaya 51951, Saudi Arabia
[2] Basic Science Department, Nile Higher Institute for Engineering and Technology, Mansoura, Egypt.
[3] Department of Mathematics and Statistics, International Islamic University, H-10, Islamabad 44000, Pakistan
[4] Department of Mathematics, College of Science and Humanities in Al-Kharj, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

* **Correspondence:** Email: renygeorge02@yahoo.com.

Abstract: The aim of this paper is to obtain the minimal rental cost of the three-phases flow shop scheduling problems. A novel strategy to tackle this issue using Pythagorean fuzzy processing time is introduced. It depends on converting the three stages machine into two stages when the minimum value of processing time of the first machine is greater than the maximum value of processing time of the second machine. The vague processing time does not convert to its crisp form. The jobs sequencing in machines is obtained using Johnson procedure. The zero element of the Pythagorean set is defined as, $\widetilde{0}^{p} = (0,1)$ i.e., it has zero membership and one nonmembership values. A numerical example include Pythagorean rental cost is delivered to demonstrate the reliability of the suggested strategy. The idle time, utilization time, and the overall cost are calculated. The idle time of all machines is zero, which minimize the required time and hence, minimize the total rental cost.

Keywords: flow shop scheduling; pythagorean fuzzy numbers; fuzzy processing time; optimization problems; decision making
Mathematics Subject Classification: 03E72, 68Q01

1. Introduction

How to schedule n jobs across m machines is referred to the flow shop scheduling problem (FSSP). Just one job at a time is processed by each machine and jobs are processed sequentially via all of the machines. Several scholars continue to study this subject [1–7]. Johnson [8] introduced the theory of FSSP. He presented a procedure for treating jobs sequencing in two or three machines with minimum overall time. Based on a bargaining game, Xing et al. [9] created feedback mechanisms for dynamic social networks. Ji et al. [10] enhancement of the overlapping community-driven feedback mechanism for social network-based group decision-making, which takes into consideration the influence of social trust connections and overlapping community detection throughout the consensus-building process.

In 1965, Zadeh proposed the concept of fuzziness [11]. Fuzzy sets are more adaptable when it comes to quantifying and getting a decision by looking at fuzzy principles. Many different types of fuzzy numbers have developed throughout the last decades [12–19]. Yeger [20] proposed the Pythagorean fuzzy set (PYFS), the most generic version of intuitionistic fuzzy set (IFS). Many authors studied PYFS and studied its applications [21–22].

Fuzzy values are utilized to depict processing times since there is a lack of clarity about them. The Fuzzy Flow Shop Scheduling (FFSS) Problem is a widely researched topic. Using a certain policy, Sathish and Ganesan [23] investigated a method to minimize the rent cost of three machines with fuzzy processing times. Khalifa [24] looked into a scheduling issue with one machine and several due dates in a fuzzy setting. A restricted FSSP was investigated by Khalifa et al. [25]. Using fuzzy due dates, they treated a multi-stage fuzzy binding strategy. Alharbi and Khalifa developed a method for handling an FSSP that involves pentagonal processing time in [26]. Al Buraikan et al. [27] introduced a novel approach to minimize processing time under pentagonal and closed interval approximation environment.

In this work, a brand-new three-stage flow shop scheduling problem with a fuzzy processing time was explored. The vague data were represented as PYFNs. The following were the study's main innovations and contributions:

1) Introducing appropriate terminologies and measurements that take potential optimum scheduling aspects into account;
2) Defining the zero element of PYFS;
3) Formulating an algorithm to choose the most efficient ordering of tasks with Pythagorean fuzzy processing times;
4) Minimizing the idle time and hence the total rental cost.

The planned study's primary goals were:

1) To minimize the machines' total processing time under the rental policy;
2) To investigate the whole study of Pythagorean fuzzy numbers (PYFNs) in the scheduling issue;
3) To express the conception of the case of optimum schedule;
4) To check out the planned study with the help of an instructive example.

The following is the breakdown of this paper. Literature review is covered in Section 2. Prelaminars related to PYFNs formulation and arithmetic operations are provided in Section 3. The suggested method is demonstrated in Section 4. In Section 5, the scheduling problem is formulated. To demonstrate the effectiveness of the suggested strategy, Section 6 provides a numerical application. Section 7 compares the obtained results with those obtained in [23,27]. Some last observations are provided in Section 8 to summarize this work.

2. Literature review

The earlier research on flow shop scheduling has been taken into account. There are several approaches involved real-time or fuzzy processing time used. A large number of articles had been published in this field. To find related articles, the title and abstract were reviewed. The proposed algorithm was devolved based on analyzing many related approaches. Table 1 lists a selection of related articles.

Table 1. Comparison of previous researches in flow shop schedulling.

Reference	Year	Research environment	No. of machines
[1]	2020	Real	Parallel-machine
[2]	2019	Real	Single-machine
[3]	2019	Real	Single-machine
[4]	2017	Real	Single-machine
[5]	2017	Stochastic	Single-machine
[6]	2020	Real	Parallel-machine
[7]	2020	Real	Two-machines
[8]	1954	Real	Two and three-machines
[23]	2012	Real	Three-machines
[24]	2020	Interval valued fuzzy	Single-machine
[25]	2021	Piecewise quadratic fuzzy	Three-machines
[26]	2021	Pentagonal fuzzy	Three-machines
[27]	2023	Pentagonal fuzzy	Three-machines
[28]	2020	Real	Multiple
[29]	2020	Real	Multiple
[30]	2020	Real	Multiple
[31]	2020	Real	Multiple
[32]	2021	Real	Multiple
[33]	2020	Real	Multiple
This study	2023	Pythagorean	Three-stages

3. Prelaminars

This section, demonstrates some essential concepts and definitions related to PYFN and its arithmetic operations.

3.1. *Some definitions related to PYFN*

Definition 1 [11]**.** A fuzzy set $\tilde{F}$ defined on $\mathbb{R}$ is a fuzzy number if its membership function (MF) $\vartheta_{\tilde{F}}(x): \mathbb{R} \rightarrow [0,1]$, satisfies:
$\vartheta_{\tilde{F}}(x)$ is an upper semi-continuous MF; $\tilde{F}$ is convex fuzzy set; and $\tilde{F}$ is normal.
Definition 2 [21]**.** Let T be a fixed set, a PYFS is the set P, such that, $P = \{\langle Y, (\alpha_P(y), \beta_P(y))\rangle : y \in Y\}$. Where, $\alpha_P(y), \beta_P(y): Y \rightarrow [0, 1]$, are the degree of membership (MD) and non- membership degree (NMD), respectively. Also, it holds that: $(\alpha_P(y))^2 + (\beta_P(y))^2 \leq 1$.

Definition 3 [19]**.** Let $\tilde{g}^P = (\alpha_n^P,\ \beta_m^P)$ and $\tilde{h}^P = (\alpha_u^P,\ \beta_v^P)$ be two PYFNs. Then, the arithmetic operations are:

(i) $\tilde{g}^P(+)\tilde{h}^P = \left(\sqrt{(\alpha_n^P)^2 + (\alpha_u^P)^2 - (\alpha_n^P)^2.(\alpha_u^P)^2}, \beta_m^P.\beta_v^P\right)$,

(ii) $\tilde{g}^P(\times)\tilde{h}^P = \left(\alpha_n^P.\alpha_u^P,\ \sqrt{(\beta_m^P)^2 + (\beta_v^P)^2 - (\beta_m^P)^2.(\beta_v^P)^2}\right)$,

(iii) $\tilde{g}^P(-)\tilde{h}^P = \begin{cases}\left(\sqrt{\frac{(\alpha_n^P)^2-(\alpha_u^P)^2}{1-(\alpha_u^P)^2}}, \frac{\beta_m^P}{\beta_v^P}\right), \alpha_n^P \geq \alpha_u^P,\ \beta_m^P \leq\ \beta_v^P, \beta_v^P \neq 0\ and \alpha_n^P \neq 1, \\ 0 \quad otherwise,\end{cases}$

(iv) $r.\tilde{g}^P = \left(\sqrt{1-(1-\alpha_n^P)^r}, (\beta_m^P)^r\right), r \succ 0.$

Definition 4 [22]**.**

(i) Score function: $SF(\tilde{g}^P) = (\alpha_n^P)^2 - (\beta_m^P)^2$.

(ii) Accuracy function: $AC(\tilde{g}^P) = (\alpha_n^P)^2 + (\beta_m^P)^2$.

Definition 5. Let$\tilde{a}^P$, and $\tilde{b}^P$ be any two PYFNs, then,

(i) $\tilde{g}^P \succ \tilde{h}^P \leftrightarrow SF(\tilde{g}^P) \succ SF(\tilde{h}^P)$,

(ii) $\tilde{g}^P \prec \tilde{h}^P \leftrightarrow SF(\tilde{g}^P) \prec SF(\tilde{h}^P)$,

(iii) $SF(\tilde{g}^P) = SF(\tilde{h}^P)$, and $AC(\tilde{g}^P) \prec AC(\tilde{h}^P) \rightarrow \tilde{g}^P \prec \tilde{h}^P$,

(iv) $SF(\tilde{g}^P) = SF(\tilde{h}^P)$, and $AC(\tilde{g}^P) \succ AC(\tilde{h}^P) \rightarrow \tilde{g}^P \succ \tilde{h}^P$,

(v) $SF(\tilde{g}^P) = SF(\tilde{h}^P)$, and $AC(\tilde{g}^P) = AC(\tilde{h}^P) \rightarrow \tilde{g}^P = \tilde{h}^P$.

The Johnson algorithm for determining the best order to complete specific tasks is described in the next section.

3.2. *Johnson's algorithm*

Johnson's rule for solving sequencing problems is demonstrated in the following steps [8]:

1) Determine the lowest processing time on the two machines.

2) (a) Process that job first if Machine 1 contains the least value.

(b) If the least value is in Machine 2, the job will be processed last.

3) (a) If the minimal time for both Machines is the same, conduct Machine 1's task first, followed by Machine 2's job.

(b) If two or more jobs in Machine 1 require the same amount of time to complete, choose the one that requires Machine 2's lowest time and do it first.

(c) Choose the work that corresponds to the minimum of Machine 1 and process it last if there is a tie for the minimum time among jobs in Machine 2.

4) Next, determine the idle time and total elapsed time 0.

In the beginning point and mathematical operations, the zero element of any set is crucial. The zero of PYFS was determined by the subsequent theorem.

Theorem 1. The zero of Pythagorean fuzzy numbers under the addition low defined in Definition 3 has zero membership value and one non membership value i.e., $\widetilde{O}^{p} = (0,1)$.

Proof. Consider any two PYFN, $\tilde{a}^{P} = (\alpha_n^P,\ \beta_m^P), \widetilde{O}^{p} = (c_n^P,\ d_m^P)$.

As $\widetilde{O}^{p}$ is the zero element, then $\tilde{a}^{P} + \widetilde{O}^{p} = \tilde{a}^{P}$,

$$\tilde{a}^{P} + \widetilde{O}^{p} = (\alpha_n^P, \beta_m^P) + (c_n^P, d_m^P) = \left(\sqrt{(\alpha_n^P)^2 + (c_n^P)^2 - (\alpha_n^P)^2.(c_n^P)^2}, \beta_m^P.d_m^P\right) = (\alpha_n^P, \beta_m^P),$$

$$\sqrt{(\alpha_n^P)^2 + (c_n^P)^2 - (\alpha_n^P)^2.(c_n^P)^2} = \alpha_n^P,$$

$$(\alpha_n^P)^2 + (c_n^P)^2 - (\alpha_n^P)^2.(c_n^P)^2 = (\alpha_n^P)^2,$$

$$(c_n^P)^2 - (\alpha_n^P)^2.(c_n^P)^2 = 0,$$

$$(c_n^P)^2(1 - (\alpha_n^P)^2) = 0,$$

$$c_n^P = 0\,, \alpha_n^P = \pm 1.$$

$$\beta_m^P.d_m^P = \beta_m^P$$

$$d_m^P = 1, \beta_m^P \neq 0.$$

$$\widetilde{O}^{p} = (c_n^P, d_m^P) = (0, 1).$$

On the other hand,

$$\tilde{a}^{P} + \widetilde{O}^{p} = (\alpha_n^P, \beta_m^P) + (0, 1) = (\alpha_n^P, \beta_m^P),$$

$$\widetilde{O}^{p} + \tilde{a}^{P} = (0,1) + (\alpha_n^P, \beta_m^P) = (\alpha_n^P, \beta_m^P),$$

$$\tilde{a}^{P} + \widetilde{O}^{p} = \widetilde{O}^{p} + \tilde{a}^{P} = \tilde{a}^{P}.$$

$$\therefore \widetilde{O}^{p} = (0, 1) \text{is the zero of PYFNs.}$$

4. Assumptions, rental policy and notations

In this section, we will introduce the considered assumptions, the rental policy and the list of the required notations.

4.1. Assumptions

Assume the following assumptions:

(i) Preemption of any job is prohibited.
(ii) One distinct task can be completed at a time.
(iii) At the beginning, all jobs are accessible.
(iv) Ignore the setting up times of machines.
(v) During the deterministic phase, every job is processed.
(vi) Due dates seem to be PYFNs.
(vii) The machines might be inactive.
(viii) The manufacturing period is unrelated to the schedule.
(ix) m operations are required for each job.
(x) Once a task is started, it must be completed.

(xi) Before processing on the second machine to follow, the first job has to be finished in the first machine.

4.2. Rental policy

The machines are rented out as required and returned as soon as they are no longer needed, i.e., the first machine will be rented out when processing jobs begins, the second machine will be rented out when the first job is finished on the first one and moved to the second machine, and the third machine will be rented out when the first job is finished on the second machine and transported.

4.3. Notations

Table 2 lists the notations that have been used.

Table 2. Notations.

Notation	Description
$\mathbf{J_k}$	Johnson procedure obtained sequence, $k = 1, \dots, m$
$\mathbf{PT_{ij}}$	Crisp processing time of task i on machine j
$\widetilde{\mathbf{PT}}_{\mathbf{ij}}$	Pythagorean processing time of task i on machine j
$\mathbf{D_j}$	Machine j
$\tilde{\mathbf{T}}_{\mathbf{ij}}(\mathbf{J_k})$	Pythagorean completion time of job i of sequence J_k
$\tilde{\boldsymbol{Z}}_{\mathbf{j}}(\mathbf{J_k})$	Pythagorean utilization time of machine D_j
$\widetilde{\mathbf{RC}}_{\mathbf{i}}$	Pythagorean rental cost of machine i
$\tilde{\mathbf{I}}_{\mathbf{ij}}(\mathbf{J_k})$	Pythagorean idle time of machine D_j for job i in the sequence (J_k)
$\widetilde{\mathbf{TC}}(\mathbf{J_k})$	Pythagorean total completion time of the jobs for sequence (J_k)
$\widetilde{\mathbf{RC}}(\mathbf{J_k})$	Pythagorean total rental cost for the sequence (J_k) of all machine

5. Problem formulation

Assume the job (i), $i = 1{:}n$,is to be served on D_j, $j = 1{:}m$ with a known rental cost $RC_i$and has a PYFN processing time$\widetilde{PT}_{ij}$ .Our goal is to find the optimal sequence $\{J_k\}$ of ordering jobs that minimize the idle time hence, the rental cost. The problem can be described as follows: Minimize $\widetilde{RC}(J_k) = \sum_{i=1}^{n}\left[\widetilde{PT}_{i1}\tilde{Z}_1 + \widetilde{PT}_{i2}\tilde{Z}_2 + \cdots + \widetilde{PT}_{im}\tilde{Z}_m\right]$, $\tilde{Z}_j$ is the utilization time of D_j, $j = 1{:}m$, subject to: the given rental policy.Assume $\widetilde{PT}_{i1}, \widetilde{PT}_{i2}, \dots, \widetilde{PT}_{im}$ be the PYFN processing times of machines $D_1, D_2, \dots, D_m$ respectively, we propose the following algorithm:

Step 1: Evaluate the associated crisp number for PYFN based on the ranking function.

Step 2: If one of the two specified requirements is met, the three machines problem can be reduced to a two machines problem:

$$\min_i \widetilde{PT}_{i1} \geq \max_i \widetilde{PT}_{ij}, j = 2{:}m-1,$$

$$\min_i \widetilde{PT}_{im} \geq \max_i \widetilde{PT}_{ij}, j = 2{:}m-1.$$

Step 3: Define two machines A_1 and A_2 such that:

$$\widetilde{A_1}_i = \sum_{j=1}^{m-1} \widetilde{PT}_{ij}\,, i = 1{:}\ n,$$

$$\widetilde{A_2}_i = \sum_{j=2}^{m} \widetilde{PT}_{ij}\,, i = 1{:}\ n,$$

where, $\widetilde{A_1}_i$, $\widetilde{A_2}_i$ are PYFN processing time of job i on A_1 and A_2 respectively.
Step 4: Evaluate the sequence $\{J_k\}$ on A_1 and A_2 using ranking method.
The above steps may be summarized in Figure 1.

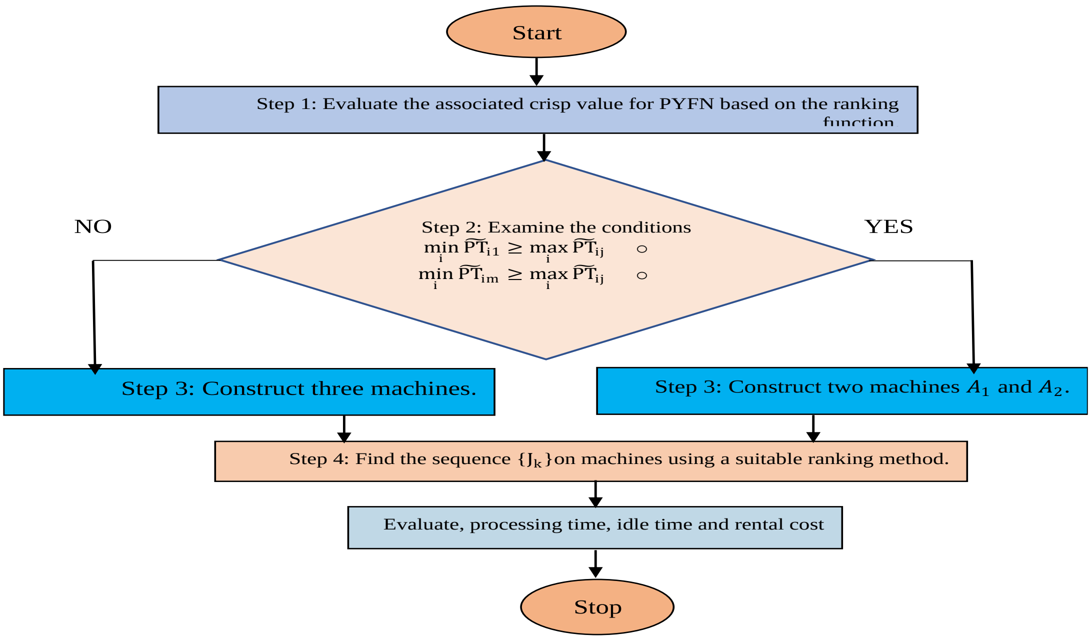

Figure 1. Flow chart of the proposed method.

6. Numerical application

Consider three machines and five tasks to be completed on them. A FSSP with PYFN processing time shown in Table 3. The rental cost of machines D_1, D_2 and D_3 per unit time are (0.9, 0.1) units, (0.8, 0.2) units and (0.7, 0.3) units respectively, under the rental policy defined in Section 3. The main objective is to minimize the total rental cost [23].

Table 3. PYFN processing time.

Jobs	PT_{i1}	PT_{i2}	PT_{i3}
1	(0.6, 0.3)	(0.6, 0.4)	(0.6, 0.5)
2	(0.7, 0.4)	(0.8, 0.7)	(0.7, 0.5)
3	(0.7, 0.2)	(0.9, 0.8)	(0.8, 0.2)
4	(0.8, 0.5)	(0.7, 0.5)	(0.8, 0.4)
5	(0.7, 0.3)	(0.6, 0.4)	(0.6, 0.2)

Evaluate the accuracy function for every $\widetilde{\mathrm{PT}}_{\mathrm{ij}}$:

$$\min_{\mathrm{i}} \widetilde{\mathrm{PT}}_{\mathrm{i1}} = (0.6, 0.3),$$

$$\max_{\mathrm{i}} \widetilde{\mathrm{PT}}_{\mathrm{i2}} = (0.7, 0.5),$$

$$\min_{\mathrm{i}} \widetilde{\mathrm{PT}}_{\mathrm{i3}} = (0.6, 0.5).$$

$$\min_{\mathrm{i}} \widetilde{\mathrm{TP}}_{\mathrm{i1}} > \max_{\mathrm{i}} \widetilde{\mathrm{TP}}_{\mathrm{i2}}.$$

Then we can reduce machines to two machines. Consider the two machines are A_1 and A_2 such that:

$$\widetilde{\mathrm{A}_1}_{\mathrm{i}} = \sum_{\mathrm{j}=1}^{2} \widetilde{\mathrm{TP}}_{\mathrm{ij}}, \qquad \widetilde{\mathrm{A}_2}_{\mathrm{i}} = \sum_{\mathrm{j}=2}^{3} \widetilde{\mathrm{TP}}_{\mathrm{ij}}$$

Table 4 gives the crisp processing times of jobs on the three machines.

Table 4. Crisp processing time.

Jobs	$\mathbf{TP_{i1}}$	$\mathbf{TP_{i2}}$	$\mathbf{TP_{i3}}$
1	0.27	0.20	0.11
2	0.33	0.15	0.24
3	0.45	0.17	0.60
4	0.39	0.24	0.48
5	0.40	0.20	0.32

Table 5 indicates the PYFN processing times and their correspondence values of score function.

Table 5. PYFN and crisp processing times of A_1 and A_2.

$\widetilde{\mathbf{A}_1}_{\mathbf{i}}$	$\boldsymbol{A_1}_{\mathbf{i}}$	$\widetilde{\mathbf{A}_2}_{\mathbf{i}}$	$\boldsymbol{A_2}_{\mathbf{i}}$	$\widetilde{\mathbf{A}_1}_{\mathbf{i}}$
1	(0.768, 0.12)	0.575	(0.768, 0.20)	0.550
2	(0.904, 0.28)	0.739	(0.904, 0.35)	0.695
3	(0.950, 0.16)	0.877	(0.965, 0.16)	0.906
4	(0.904, 0.25)	0.755	(0.904, 0.20)	0.777
5	(0.821, 0.12)	0.660	(0.768, 0.08)	0.583

The optimal sequence obtained by Johnson's Algorithm [8], is $4 \to 3 \to 2 \to 5 \to 1.$
From Table 6, we can find that:

- The required time to complete all jobs, $\widetilde{\mathrm{TC}}(\mathrm{J_k}) = (0.999, 0.0008)$,
- Idle time of $\mathrm{D_1}$ is, $\tilde{\mathrm{I}}_1 = (0, 1)$,
- Idle time of $\mathrm{D_2}$ is, $\tilde{\mathrm{I}}_2 = (0, 1)$,
- Idle time of $\mathrm{D_3}$ is, $\tilde{\mathrm{I}}_3 = (0, 1)$.

The utilization time of machines:

$\tilde{Z}_1 = (0.985, 0.0036) - (0, 1) = (0.985, 0.0036)$ hrs.

$\tilde{Z}_2 = (0.997, 0.008) - (0.8, 0.5) = (0.991, 0.016)$ hrs.

$\tilde{Z}_3 = (0.999, 0.0008) - (0.904, 0.25) = (0.994, 0.0032)$ hrs

The machines rental cost,

$\widetilde{RC}_1 = (0.9,\ 0.1) * (0.985, 0.0036) = (0.8865, 0.1)$ units.

$\widetilde{RC}_2 = (0.8, 0.2) * (0.991, 0.016) = (0.7928, 0.2)$ units.

$\widetilde{RC}_3 = (0.7,\ 0.3) * (0.994, 0.0032) = (0.6958, 0.3)$ units.

Total rental cost,

$\widetilde{RC}(J_k)=\sum_{j=1}^{3} \widetilde{RC}_j$= (0.9792, 0.006) units.

Table 6. Time in and time out.

	D_1		D_2		D_3	
Job	Time in	Time out	Time in	Time out	Time in	Time out
4	(0, 1)	(0.8, 0.5)	(0.8, 0.5)	(0.904, 0.25)	(0.904, 0.25)	(0.910, 0.1)
3	(0.8, 0.5)	(0.904, 0.1)	(0.904, 0.1)	(0.982, 0.08)	(0.982, 0.08)	(0.994, 0.016)
2	(0.904, 0.1)	(0.952, 04)	(0.982, 0.08)	(0.994, 0.056)	(0.994, 0.016)	(0.997, 0.008)
5	(0.952, 04)	(0.976, 0.012)	(0.994, 0.056)	(0.996,0.0224)	(0.997, 0.008)	(0.998, 0.0016)
1	(0.976, 0.012)	(0.985, 0.0036)	(0.996, 0.0224)	(0.997, 0.008)	(0.998, 0.0016)	(0.999, 0.0008)

Table 7 provides a summary of the aforementioned findings:

Table 7. Utilization time, idle time and machines rental cost.

Item	D_1	D_2	D_3
Idle time	$(0, 1)$	$(0, 1)$	$(0, 1)$
Utilization time	$(0.985, 0.0036)$	$(0.991, 0.016)$	$(0.994, 0.0032)$
Rental cost	$(0.8865, 0.1)$	$(0.7928, 0.2)$	$(0.6958, 0.3)$

7. Comparative study

In this section, a comparative study with some existing studies will be introduced. The results with different fuzzy environments will be compared, so, the associated crisp values will be used based on the score function of each fuzzy number defined in the associated article.

Table 8. Comparing processing time.

Reference	Type of fuzzy number	Processing time	Crisp value of processing time
[23]	Triangular	(61, 63, 65)	63
[27]	Pentagonal	(61, 62, 63, 64, 65)	63
Proposed algorithm	Pythagorean	$(0.999, 0.0008)$	0.998

Table 9. Total renal cost.

Reference	Environment	Rental cost type
[23]	Triangular	Crisp
[27]	Pentagonal	Crisp
Proposed algorithm	Pythagorean	Pythagorean

Table 10. Idle time of machines.

Reference	D_1	D_2	D_3
[23]	(9, 11, 13)	(16, 18, 20)	(7, 9, 11)
[27]	$(-3, 4,11,18,25)$	$(-13, 2.5,18,33.5,49)$	$(-24, -7.5,9,25.5,42)$
Proposed algorithm	(0, 1)	(0, 1)	(0, 1)

From the above Tables 8–10, it is observed that:

1) In [23,27] the rental costs of machines are crisp values, while in the proposed method the machines rental costs are PYFNs.
2) The idle time in the proposed algorithm is (0, 1) i.e., zero, while the idle times in [23,27] non-zero.
3) When minimizing machine idle time, then the total rental cost will be minimized also.

8. Conclusions

A technique for solving the particular structured three-stage FFSS with PYFN processing time is presented. The overall rental cost of machines is calculated. Using Johansson's procedure, the sequence of treating jobs is evaluated. The zero element of the PYFS is defined as (0, 1) and used as a starting point in Table 6. Under the given policy and the proposed method, the idle time of all machines is zero, that minimizing the total completion time, hence the rental cost. The introduced approach is more efficient as minimize the total time of completing jobs and the required rental cost.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

Conflict of interest

The authors declare no conflict of interest.

References

1. M. S. S. Mir, J. Rezaeian, H. Mohamadian, Scheduling parallel machine problem under general effects of deterioration and learning with past-sequence-dependent setup time: heuristic and meta-heuristic approaches, *Soft Comput.*, **24** (2020), 1335–1355. https://doi.org/10.1007/s00500-019-03970-z
2. W. Luo, Y. Xu, W. Tong, G. Lin, Single-machine scheduling with job-dependent machine deterioration, *J. Sched.*, **22** (2019), 691–707. https://doi.org/10.1007/s10951-019-00622-w
3. R. Chen, J. Yuan, Unary NP-hardness of single-machine scheduling to minimize the total tardiness with deadlines, *J. Sched.*, **22** (2019), 595–601. https://doi.org/10.1007/s10951-019-00615-9
4. W. Luo, F. Liu, On single-machine scheduling with workload-dependent maintenance duration, *Omega*, **68** (2017), 119–122. https://doi.org/10.1016/j.omega.2016.06.008
5. S. Atakan, B. Kerem, N. Noyan, Minimizing value-at-risk in single-machine scheduling, *Ann. Oper. Res.*, **248** (2017), 25–73. https://doi.org/10.1007/s10479-016-2251-z
6. L. Zhang, O. Deng, G. Gong, W. Han, A new unrelated parallel machine scheduling problem with tool changes to minimize the total energy consumption, *Int. J. Prod. Res.*, **58** (2020), 6826–6845. https://doi.org/10.1080/00207543.2019.1685708
7. J. Xu, S. Zhang, Y. Hu, Research on construction and application for the model of multistage job shop scheduling problem, *Math. Probl. Eng.*, **2020** (2020), 635739. https://doi.org/10.1155/2020/6357394
8. S. M. Johnson, Optimal two and three stages production schedule with set up times included, *Naval Research Logistics Quarterly*, **1** (1954), 61–68. https://doi.org/10.1002/nav.3800010110
9. Y. Xing, J. Wu, F. Chiclana, G. Yu, M. Cao, E. Herrera-Viedma, A bargaining game-based feedback mechanism to support consensus in dynamic social network group decision making, *Inform. Fusion*, **93** (2023), 363–382. https://doi.org/10.1016/j.inffus.2023.01.004
10. F. Ji, F. J. Wu, F. Chiclana, S. Wang, H. Fujita, E. Herrera-Viedma, The overlapping community driven feedback mechanism to support consensus in social network group decision making, *IEEE T. Fuzzy Syst.*, **2023** (2023), 1–15. https://doi.org/10.1109/TFUZZ.2023.3241062
11. L. A. Zadeh, Fuzzy Sets, *Inf. Control*, **8** (1996), 338–353.
12. H. Garg, S. A. Edalatpanah, S. El-Morsy, H. A. Khalifa, On stability of continuous cooperative static games with possibilistic parameters in the objective functions, *Comput. Intel. Neurosc.*, **2022** (2022), 6979075. https://doi.org/10.1155/2022/6979075
13. M. Saeed, M. Ahsan, M. H. Saeed, A. Mehamood, S. El-Morsy, Assessment of solid waste management strategies using an efficient complex fuzzy hypersoft set algorithm based on entropy and similarity measures, *IEEE Access*, **9** (2021), 150700–150714. https://doi.org/10.1109/ACCESS.2021.3125727
14. A. U. Rahman, M. Saeed, M. Arshad, S. El-Morsy, Multi-attribute decision-support system based on aggregations of interval-valued complex neutrosophic hypersoft set, *Appl. Comput. Intell. S.*, **2021** (2021), 4368770. https://doi.org/10.1155/2021/4368770
15. S. A. El-Morsy, Optimization of fuzzy zero-base budgeting, *Computational Algorithms and Numerical Dimensions*, **1** (2022), 147–154.
16. M. Saeed, M. Ahsan, M. H. Saeed, S. El-Morsy, An optimized complex fuzzy hypersoft set system based approach for the evaluation of strategic procurement techniques for fuel cell and hydrogen components, *IEEE Access*, **10** (2022), 71612–71631. https://doi.org/10.1109/ACCESS.2022.3188293

17. M. Ihsan, M. Saeed, A. U. Rahman, H. A. Khalifa, S. El-Morsy, An intelligent fuzzy parameterized multi-critera decision-support system based on intuitionistic fuzzy hypersoft expert set for automobile evolution, *Adv. Mech. Eng.*, **14** (2022), 1–20. https://doi.org/10.1177/16878132221110005
18. H. A. Khalifa, M. Saeed, A. U. Rahman, S. El-Morsy, An application of pentagonal neutrosophic linear programming for stock portfolio optimization, *Neutrosophic Sets Sy.*, **51** (2022), 653–665.
19. K. Rathi, S. Balamohan, Representation and ranking of fuzzy numbers with heptagonal membership function using value and ambiguity index, *Applied Mathematical Sci*ences, **8** (2014), 4309–4321. https://doi.org/10.12988/AMS.2014.45363
20. R. R. Yager, Pythagorean membership grades in multi-criteria decision making, *IEEE T. Fuzzy Syst.*, **22** (2014), 958–965. https://doi.org/10.1109/TFUZZ.2013.2278989
21. R. M. Zulqarnain, I. Siddique, S. El-Morsy, Einstein-ordered weighted geometric operator for pythagorean fuzzy soft set with its application to solve MAGDM problem, *Math. Probl. Eng.*, **2022** (2022), 5199427. https://doi.org/10.1155/2022/5199427
22. X. J. Gou, Z. S. Xu, P. J. Ren, The properties of continuous pythagorean fuzzy information, *Int. J. Intell. Syst.*, **31** (2016), 401–424. https://doi.org/10.1002/int.21788
23. S. Sathish, K. Ganesan, Flow shop scheduling problem to minimize the rental cost under fuzzy environment, *Journal of Natural Sciences Research*, **2** (2012), 62–68.
24. H. Khalifa, On single machine scheduling problem with distinct due dates under fuzzy environment, *International Journal of Supply and Operations Management*, **7** (2020), 272–278. https://doi.org/10.22034/IJSOM.2020.3.5
25. H. A. E. Khalifa, S. S. Alodhaibi, P. Kumar, Solving constrained flow-shop scheduling problem through multistage fuzzy binding approach with fuzzy due dates, *Adv. Fuzzy Syst.*, **2021** (2021), 6697060. https://doi.org/10.1155/2021/6697060
26. M. G. Alharbi, H. A. E. Khalifa, On a flow-shop scheduling problem with fuzzy pentagonal processing time, *J. Math.*, **2021** (2021), 6695174. https://doi.org/10.1155/2021/6695174
27. A. Alburaikan, H. Garg, H. A. E. Khalifa, A novel approach for minimizing processing times of three-stage flow shop scheduling problems under fuzziness, *Symmetry*, **15** (2023), 130. https://doi.org/10.3390/sym15010130
28. K. Shaukat, S. Luo, V. Varadharjan, A novel deep learning-based approach for malware detection, *Eng. Appl. Artif. Intel.*, **122** (2023), 106030. https://doi.org/10.1016/j.engappai.2023.106030
29. K. Shaukat, S. Luo, V. Varadharjan, I. A. Hameed, M. Xu, A survey on machine learning tchniques for cyber security in the last decade, *IEEE Access*, **8** (2020), 222310–222354. https://doi.org/10.1109/ACCESS.2020.3041951
30. U. Javed, K. Shaukat, I. A. Hameed, F. Iqbal, T. M. Alam, S. Luo, A review of content-based and context-based recommendation systems, *Int. J. Emerg. Technol.*, **16** (2021), 274–306. https://doi.org/10.3991/ijet.v16i03.18851
31. T. M. Alam, K. Shaukat, I. A. Hameed, S. Luo, M. Umer, S. Shabbiri, et al., An investigation of credit card default prediction in the imbalanced datasets, *IEEE Access*, **8** (2020), 201173–201198. https://doi.org/10.1109/ACCESS.2020.3033784
32. T. M. Alam, K. Shaukat, M. Mushtaq, Y. Ali, M. Khushi, S. Luo, et al., Corporate bankruptcy prediction: an approach towards better corporate world, *The Computer Journal*, **64** (2021), 1731–1746. https://doi.org/10.1093/comjnl/bxaa056

33. K. Shaukat, S. Luo, S. Chen, D. Liu, Cyber threat detection using machine learning techniques: a performance evaluation perspective, *International Conference on Cyber Warfare and Security (ICCWS)*, Islamabad, Pakistan, 2020, 1–6. https://doi.org/10.1109/ICCWS48432.2020.9292388