JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-21, No.-05, May (2026) pp 162-172 ISSN (Print) 0973-8975

HYBRID DECISION-MAKING IN FLOW SHOP SCHEDULING: CONTRASTING BB AND NEH WITH INTERVAL VALUED INTUITIONISTIC FUZZY DATA

Rajvinder Kaur[1], Deepak Gupta[2], Sonia Goel[3]

[1,2,3] Department of Mathematics, Maharishi Markandeshwar Engineering College, Maharishi Markandeshwar (Deemed to be University), Mullana, Haryana, India.

Email: [1]rajvinderk.modicollege@gmail.com, [2]guptadeepak@yahoo.co.in, [3]sonia.mangla14@gmail.com

Corresponding Author: **Sonia Goel**

https://doi.org/10.26782/jmcms.2026.05.00011

(Received: February 11, 2026; Revised: May 05, 2026; Accepted : May 12, 2026)

Abstract

Scheduling problems represent a core challenge in the efficient management of industrial and service operations. Due to their structural complexity and significant practical relevance in both manufacturing and service sectors, Hybrid Flow Shop Scheduling Problems (HFSSPs) are widely recognized as NP-hard. Scheduling in contemporary manufacturing and production systems sometimes entails ambiguous and uncertain information, rendering classical deterministic methods less efficacious. This work presents a novel comparative analysis of the exact method Branch and Bound (BB) and heuristic algorithm Nawaz, Enscore, and Ham (NEH) for addressing the hybrid flow shop scheduling problem (HFSSP), where processing times are articulated via Interval-Valued Intuitionistic Fuzzy Sets (IVIFS). A ranking and scoring algorithm is utilised to convert IVIFS data into computationally manageable values, facilitating integration with BB and NEH methodologies. The results offer valuable insights for scheduling in uncertain and imprecise production environments, demonstrating how hybrid decision-making strategies that combine exact and heuristic methods can lead to more effective solutions.

Keywords: Interval-Valued Intuitionistic Fuzzy Sets, Hybrid Flow Shop Scheduling, Score Function, Makespan

I. Introduction

Scheduling continues to be a key issue in production and operations management, as it strongly influences resource utilization, operational efficiency, and overall system performance. A basic class of problems is permutation scheduling, where jobs follow a uniform processing order among processors. Hybrid flow shops (HFS) are prevalent manufacturing settings where a collection of n jobs is handled through m phases to optimise a specified target function. Hybrid flow shop scheduling (HFSS) has gained considerable attention in recent years due to its wide applicability in complex production and service systems, such as electronics, semiconductor manufacturing, textile processing, and cloud computing [XIV]. A further complication in real-world scheduling arises from uncertainty in processing times, which may result from fluctuating machine conditions, operator variability, or incomplete information. Conventional deterministic models frequently fall short when dealing with real-world uncertainty. To address this shortcoming, fuzzy set theory has emerged as a powerful and widely used modelling approach. The integration of Interval-Valued Intuitionistic Fuzzy Sets (IVIFS) provides a robust mathematical framework to handle this ambiguity by capturing both the degree of belongingness (membership) and non-belongingness (non-membership) as intervals, rather than single points. IVIFS provide a more advanced method for managing ambiguity in data [II]. Numerous accurate approaches, heuristic methods, and metaheuristics have been utilised in HFSS. This study aims to systematically compare BB and NEH using IVIFS for processing times. BB employs lower bound estimates to eliminate branches that cannot yield an optimal solution. NEH ranks jobs based on total processing time and builds a sequence by inserting jobs one by one into the best possible positions.

II. Literature Survey

The fundamental flow shop scheduling concept was initially examined thoroughly by Johnson in 1954[VII]. Research on HFSS has been comprehensive, with initial research concentrating on deterministic models. Dessouky et al. [III] are the pioneers in utilising BB approaches for two and three-stage HFS. Linn and Zhang [X] presented a comprehensive survey of HFSS, classifying solution methods into exact, heuristic, and metaheuristic approaches. Lee and Kim [IX] introduced the BB strategy aimed at minimising total delay. Classical methods such as BB have been applied to small-scale HFSS problems, providing exact solutions but suffering from combinatorial explosion [XIII]. Gupta and Goel [VI] solved a two-stage HFSS using the BB method. Malhotra et al. [XI] presented a comparative analysis of BB with a metaheuristic approach on a three-stage HFSS. Heuristic approaches, particularly the NEH heuristic [XII], have proven highly effective for permutation flow shop scheduling (FSS) and have inspired extensions to hybrid environments. Kurniawan and Farizal [VIII] devised a novel approach for addressing FSS problems, grounded in the principles of the CDS and NEH methodologies, namely NEHLPD, NEHLPD1, and NEHLPD2.To address uncertainty, fuzzy sets have been incorporated into scheduling models. Dubois and Prade [IV] were among the first to introduce fuzzy set theory into decision-making problems. IVIFS further advanced this representation by modelling uncertainty through intervals rather than precise values, providing greater flexibility [I, II]. Researchers like Gholami et al. [V] have explored HFS problems under fuzzy conditions. Applications of IVIFS in scheduling remain relatively limited, though recent studies have begun to explore their potential. Senapati et al. [XV] investigated aggregation operators on IVIFS. However, few works have systematically compared exact and heuristic methods in HFSS under IVIFS-based processing times. This gap motivates the present study, which contrasts the BB and NEH heuristics in the HFS environment, using IVIFS to represent uncertain processing times. The comparison contributes both to the theoretical development of scheduling models involving uncertainty and to practical decision-making for complex manufacturing systems.

III. Practical Application

In a courier parcel processing centre, each parcel undergoes two sequential processing stages as shown in fig. 1. In the first stage, parcels are processed on one of several identical scanning machines for barcode reading and validation, followed by a second stage where all parcels are handled by a single packaging station for sealing and labelling. This setup corresponds to a two-stage HFSSP, where several parallel machines operate in the initial stage, followed by a single machine in the second stage. In real operational environments, processing times and operating costs are highly uncertain due to variations in barcode quality, parcel size, equipment performance, and operator behaviour. Such uncertainty makes deterministic modelling inadequate. Therefore, processing parameters are represented using interval-valued intuitionistic fuzzy numbers, IVIFN which effectively capture imprecision, hesitation, and partial knowledge. This framework allows decision makers to capture imprecision more effectively than classical deterministic or fuzzy models. The scheduling purpose is to ascertain an optimal or near-optimal parcel sequence that minimises makespan under fuzzy conditions.

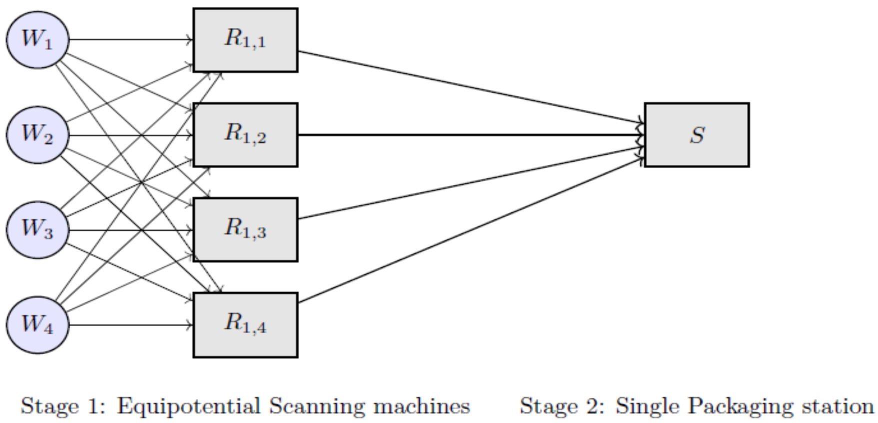

Fig. 1. Two-stage HFSS problem

IV. Preliminaries

Assume that $\mathcal{E}$ is a recognized universe of discourse set such that

$\mathcal{E} = \{e_1, e_2, \dots, e_n\}$.

Definition 4.1[XV] [XVI]

Interval-valued intuitionistic fuzzy sets (IVIFS) in $\mathcal{E}$ are given by

$\mathcal{F} = \{\langle e, \beta_{\mathcal{F}}(e), \delta_{\mathcal{F}}(e)\rangle : e \in \mathcal{E}\}$, Where $\beta_{\mathcal{F}}(e): \mathcal{E} \to S[0,1]$, $\delta_{\mathcal{F}}(e): \mathcal{E} \to S[0,1]$ and $S[0,1]$ is the set of all subintervals of $[0,1]$.$\beta_{\mathcal{F}}(e)$ and $\delta_{\mathcal{F}}(e)$ denotes intervals of degree of membership function and degree of non-membership function of the elements e in $\mathcal{E}$, $\beta_{\mathcal{F}}(e) = [\beta_{\mathcal{F}}{}^{L}(e), \beta_{\mathcal{F}}{}^{U}(e)]$, $\delta_{\mathcal{F}}(e) = [\delta_{\mathcal{F}}{}^{L}(e), \delta_{\mathcal{F}}{}^{U}(e)]$ for all $e \in$

$\mathcal{E}$, including the condition that, $0 \leq \beta_{\mathcal{F}}{}^{U}(e) + \delta_{\mathcal{F}}{}^{U}(e) \leq 1$. $\pi_{\mathcal{F}}(e) = [\pi_{\mathcal{F}}{}^{L}(e), \pi_{\mathcal{F}}{}^{U}(e)]$ denotes the hesitancy degree of the element of e in $\mathcal{F}$, where $\pi_{\mathcal{F}}{}^{L}(e) = 1 - \beta_{\mathcal{F}}{}^{U}(e) - \delta_{\mathcal{F}}{}^{U}(e)$ and $\pi_{\mathcal{F}}{}^{U}(e) = 1 - \beta_{\mathcal{F}}{}^{L}(e) - \delta_{\mathcal{F}}{}^{L}(e)$.

For convenience, $\mathcal{M} = \left(\left[\beta_{\mathcal{M}}{}^{L}, \beta_{\mathcal{M}}{}^{U}\right], \left[\delta_{\mathcal{M}}{}^{L}, \delta_{\mathcal{M}}{}^{U}\right]\right)$ is called IVIFN, where $\left[\beta_{\mathcal{M}}{}^{L}, \beta_{\mathcal{M}}{}^{U}\right] \in S[0,1]$ and $\left[\delta_{\mathcal{M}}{}^{L}, \delta_{\mathcal{M}}{}^{U}\right] \in S[0,1]$

Definition 4.2[XV] [XVI]

For any IVIFN $\mathcal{M} = \left(\left[\beta_{\mathcal{M}}{}^{L}, \beta_{\mathcal{M}}{}^{U}\right], \left[\delta_{\mathcal{M}}{}^{L}, \delta_{\mathcal{M}}{}^{U}\right]\right)$, the score function $\mathrm{sco}(\mathcal{M})$, accuracy function $\mathrm{acc}(\mathcal{M})$ of $\mathcal{M}$ is defined as follows:

$$\mathrm{sco}(\mathcal{M}) = \frac{1}{2}\left(\beta_{\mathcal{M}}{}^{L} + \beta_{\mathcal{M}}{}^{U} - \delta_{\mathcal{M}}{}^{L} - \delta_{\mathcal{M}}{}^{U}\right)$$
$$\mathrm{acc}(\mathcal{M}) = \frac{1}{2}\left(\beta_{\mathcal{M}}{}^{L} + \beta_{\mathcal{M}}{}^{U} + \delta_{\mathcal{M}}{}^{L} + \delta_{\mathcal{M}}{}^{U}\right)$$

The score and accuracy functions preserve dominance, i.e., if $\mathcal{M}$ dominates $\mathcal{N}$, then $\mathrm{sco}(\mathcal{M}) \geq \mathrm{sco}(\mathcal{N})$ and if $\mathrm{sco}(\mathcal{M}) = \mathrm{sco}(\mathcal{N})$ then $\mathrm{acc}(\mathcal{M}) \geq \mathrm{acc}(\mathcal{N})$.

The score function is monotonic, increasing with membership values and decreasing with non-membership values.

V. Mathematical Formulation

This problem describes a two-stage mathematical model in which the first stage has m equipotential machines and the second stage has a single machine, where operating costs, processing times, etc., are represented as IVIFN.

V.i. Notations

- K: Total jobs handled by machines R and S.
- W_i: ith job
- m: Number of equipotential machines at first stage
- R_i (i=1,2,…,m): Equipotential machine of type R
- $\mathcal{C}_{ij}, \mathcal{C}_{ij}{}^{I}$: Crisp and IVIFN operating cost of job i on machine j.
- $\mathcal{P}_{ij}, P_{ij}{}^{I}$: Crisp and IVIFN processing times of job i on machine j.
- $t_i, t_i{}^{I}$: Total crisp and IVIFN available time for first stage machines R_i.
- Ţ_{ij}: Processing time of the ith job allotted to R_j equipotential machines.

V.ii. Assumptions

1. All first-stage equipotential devices are available at time zero and have different consumption costs.
2. Not every job must be completed by every parallel machine.
3. Equipotential machines of stage one may have the same opening time for processing.
4. It is presumed that jobs transition directly from one machine to another.
5. Jobs are not subject to pausing, halting, or interruption.

V.iii. Mathematical Model

This model consists of K (W_i, i=1,2,…, K) jobs, a 2-machine (R, S) problem, in which the first stage has m machines (R_i, i=1,2,…,m) and the second stage has a single machine S as presented in Table 1.

Table 1: Structure of Mathematical Model

Jobs		Machine R				Processing Time (Machine R)	Processing Time (Machine S)
W_i	R_1	R_2	R_3	…	R_m	$\mathcal{P}_{i1}^I$	$\mathcal{P}_{i2}^I$
W_1	$\mathcal{C}_{11}^I$	$\mathcal{C}_{12}^I$	$\mathcal{C}_{13}^I$	…	$\mathcal{C}_{1m}^I$	$\mathcal{P}_{11}^I$	$\mathcal{P}_{12}^I$
W_2	$\mathcal{C}_{21}^I$	$\mathcal{C}_{22}^I$	$\mathcal{C}_{23}^I$	…	$\mathcal{C}_{2m}^I$	$\mathcal{P}_{21}^I$	$\mathcal{P}_{22}^I$
W_3	$\mathcal{C}_{31}^I$	$\mathcal{C}_{32}^I$	$\mathcal{C}_{33}^I$	…	$\mathcal{C}_{3m}^I$	$\mathcal{P}_{31}^I$	$\mathcal{P}_{32}^I$
.	.	.	.	…	.	.	.
.	.	.	.	…	.	.	.
.	.	.	.	…	.	.	.
W_K	$\mathcal{C}_{K1}^I$	$\mathcal{C}_{K2}^I$	$\mathcal{C}_{K3}^I$	…	$\mathcal{C}_{Km}^I$	$\mathcal{P}_{K1}^I$	$\mathcal{P}_{K2}^I$
Available Times→	t_1^I	t_2^I	t_3^I	…	t_m^I		

VI. Proposed Methodology

Step 1: First convert processing times $\mathcal{P}_{ij}^I(i = 1,2, \dots, K; j = 1,2)$, operating costs $\mathcal{C}_{ij}^I(i = 1,2, \dots, K; j = 1,2, \dots, m)$ and available times $t_j^I, (j = 1,2, \dots, m)$into crisp values using the score function sco of IVIFN.

Step 2: Determine if the transportation problem is balanced by checking $\sum_{j=1}^{m} t_j = \sum_{i=1}^{K} P_{i1}$; if not, insert a fake row or column to achieve balance.
Step 3: For the balanced transportation issue, derive the initial viable solution utilizing Vogel's Approximation Method (VAM).

Step 4: Calculate the optimal sequence of jobs using the BB Method as

(a) Start with an empty schedule and use the formula given below

$$k_1 = max\left(\sum_{i=1}^{K} \mathrm{T}_{ij}\right)_{j=1,2,\dots,m} + min_{i \in W_t'}(\mathcal{P}_{i2});$$

$$k_2 = \left\{max\left(\mathrm{T}_{ij}\right)_{i \in W_t}\right\}_{j=1,2,\dots,m} + \sum_{i=1}^{K} \mathcal{P}_{i2} ;$$

Where W_t denote jobs of branching tree and W_t' denote jobs which are not considered under the branching tree.
Find $L = max\{k_1, k_2\}$.

(b) Derive lower limits for task permutations and determine the minimal value from this set of bounds.

(c) Create branches by adding jobs sequentially and constrain each node using a lower bound. Eliminate nodes with inferior limits. The process continues until all nodes have been examined or eliminated, after which the most efficient timetable is obtained.
(d) Determine the In-out Table for the acquired work sequence and calculate the makespan for the job sequence.

VII. Numerical Examples

Consider a two-stage hybrid FSSP with processing times, Operating costs, and available times of both stages represented as IVIFN as in table-2.

Table 2: Input matrix

	R_1	R_2	R_3	R_4	R	S
W_1	([0.63, 0.8], [0.12, 0.17])	([0.41, 0.42], [0.1, 0.11])	([0.44, 0.84], [0.16, 0.17])	([0.44, 0.7], [0.12, 0.25])	([0.55, 0.7], [0.01, 0.25]),	([0.53, 0.77], [0.02, 0.07])
W_2	([0.56, 0.58] , [0.2, 0.32])	([0.44, 0.56], [0.16, 0.43])	([0.63, 0.85], [0.02, 0.13])	([0.69, 0.77], [0.09, 0.15])	([0.59, 0.78], [0.1, 0.3])	([0.68, 0.84], [0.08, 0.16])
W_3	([0.52, 0.71], [0.12, 0.24])	([0.49, 0.89], [0.14, 0.16])	([0.42, 0.86], [0.08, 0.13])	([0.57, 0.7], [0.09, 0.26])	([0.42, 0.45], [0.01, 0.39])	([0.46, 0.55], [0.02, 0.36])
W_4	([0.65, 0.82], [0.16, 0.18])	([0.58, 0.67], [0.18, 0.19]),	([0.58, 0.76], [0.14, 0.18])	([0.66, 0.68], [0.03, 0.21])	([0.58, 0.84], [0.0, 0.01])	([0.56, 0.68], [0.09, 0.31])
	([0.58, 0.84], [0.0, 0.01])	([0.42, 0.45], [0.01, 0.39])	([0.59, 0.78], [0.1, 0.3])	([0.55, 0.7], [0.01, 0.25]),		

VII.i. Solution by Proposed Algorithm

After applying the score function sco to the input matrix, table 3 is formed, and the distribution after VAM is shown in table-4.

Table 3: Matrix after applying score function sco changed

	R_1	R_2	R_3	R_4	R	S
W_1	0.57	0.31	0.475	0.385	0.495	0.605
W_2	0.31	0.205	0.665	0.61	0.485	0.64
W_3	0.435	0.54	0.535	0.46	0.235	0.315
W_4	0.565	0.44	0.51	0.55	0.705	0.42
	0.705	0.235	0.485	0.495		

Table 4: Distribution of processing times on stage -1 equipotential machines using VAM

	R_1	R_2	R_3	R_4	R	S
W_1	0	0.235	0	0.26	0.495	0.605
W_2	0.485	0	0	0	0.485	0.64
W_3	0.22	0	0	0.015	0.235	0.315
W_4	0	0	0.485	0.22	0.705	0.42

Now apply the proposed technique to find the optimum sequence and find the lower bounds of jobs as presented in Table 5.

Table 5: Depicting the lower bound of jobs

i	k_1	k_2	L(i)=max{k_1, k_2}
1	0.705+0.315=1.02	0.26+1.98=2.24	2.24
2	0.705+0.315=1.02	0.485+1.98=2.465	2.465
3	0.705+0.42=1.125	0.22+1.98=2.2	2.2
4	0.705+0.315=1.02	0.485+1.98=2.465	2.465

The lower bound for the first branch is 2.2, corresponding to job 3. Insert job 3 in the first location and determine the lower bounds for the second branch as illustrated in Table 6.

Table 6: Lower Bound Evaluation for the Partial Sequence with Job 3 Fixed First.

i	k_1	k_2	L(i)=max{k_1, k_2}
{3,1}	0.705+0.42=1.125	0.275+1.98=2.255	2.255
{3,2}	0.705+0.42=1.125	0.705+1.98=2.685	2.685
{3,4}	0.705+0.605=1.31	0.485+1.98=2.465	2.465

Lower bound for the second branch is 2.255, and it corresponds to the subsequence {3,1}. Fix job 1 at the second position and move for the third branch lower bounds as shown in Table 7.

Table 7: Lower Bound Values with Job Sequence {3, 1} Fixed

i	k_1	k_2	L(i)=max{k_1, k_2}
{3,1,2}	0.705+0.42=1.125	0.705+1.98=2.685	2.685
{3,1,4}	0.705+0.64=1.345	0.495+1.98=2.475	2.475

Lower bound for the third branch is 2.475 and corresponds to the subsequence {3,1,4}. So the final optimum sequence is {3,1,4,2}, and the in-out table is presented in table-8.

Table 8: In-out Table for optimum sequence {3, 1, 4, 2}

	R_1	R_2	R_3	R_4	R
W_3	0-0.22	-	-	0-0.015	0.22-0.535
W_1	-	0-0.235	-	0.015-0.275	0.535-1.14
W_4	-	-	0-0.485	0.275-0.495	1.14-1.56
W_2	0.22-0.705	-	-	-	1.56-2.2

VII.ii. Solution by the modified NEH method

The NEH method [XII] is a heuristic technique utilised to enhance processing efficiency in FSSPs. This method efficiently establishes the work sequence to reduce total processing time. The NEH approach is famous for its simplicity and effectiveness. The established procedure of the NEH is modified for HFSSP as follows:

1. Repeat Steps 1–3 of the proposed methodology using the input matrix presented in Table 2 to obtain the results shown in Table 4.
2. For ranking of jobs, the processing times of each job across all stages are summed to obtain the total processing time.
3. The jobs are then arranged in decreasing order of their total processing times.
4. From this ordered list, the first two jobs are selected, and the better of the two possible sequences is identified.
5. Each remaining job is subsequently inserted into every possible position within the current best sequence. The position that results in the smallest partial makespan is chosen, and the job is fixed in that position.
6. This procedure is repeated until all jobs are scheduled, yielding a near-optimal sequence; the makespan of this sequence is then obtained from Table 4.

Consider Table 2 as the input table and apply the above two steps to the input table to get Table 4 and Table 9.

Table 9: Processing Times of all jobs

	R	S	Sum of Processing Times
W_1	0.495	0.605	1.1
W_2	0.485	0.64	1.125
W_3	0.235	0.315	0.55
W_4	0.705	0.42	1.125

Arrange Jobs in descending order according to their sum of processing times: 2<4<1<3. Choose the first two jobs in sequence; these jobs can be arranged as{2,4}or{4,2}. Makespan of subsequence {2,4} equals 1.61, and Makespan of subsequence {4,2} equals 1.83. Minimum make span is 1.61 corresponding to sequence{2,4}. Now, the next job in sequence is 1, placing this job in the already selected sequence{2,4} gives rise to three new possibilities: {2,4,1},{1,2,4}, or {2,1,4}. Make span of these sequences are 2.29, 2.16, and 2.15, respectively. Select sequence {2,1,4} and the left job is 3 .Try to place job 3 in the already selected sequence {2,1,4}, which gives rise to four sequences {3,2,1,4},{2,3,1,4},{2,1,3,4}, and {2,4,1,3} whose make spans are equal to 2.385, 2.465, 2.465, and 2.605, respectively. So, a near-optimal job sequence is {3,2,1,4}, and the in-out table for the sequence is presented in Table 10.

Table 10: In-out Table for optimum sequence {3, 2, 1, 4}

	R_1	R_2	R_3	R_4	R
W_3	0-0.22	-	-	0-0.015	0.22-0.535
W_2	0.22-0.705	-	-	-	0.705-1.345
W_1	-	0-0.235	-	0.015-0.275	1.345-1.95
W_4	-	-	0-0.485	0.275-0.495	1.95-2.37

VIII. Comparative Performance Analysis

By repeating the above procedure, additional numerical experiments involving different numbers of jobs were conducted using both the proposed methodology and the modified NEH algorithm. The comparative results are presented in Table 11, and the corresponding graphical comparison is illustrated in Fig. 2.

Table 11: Comparison between Proposed Algorithm and Modified NEH

No of jobs	Proposed Methodology	Modified NEH
3	1.94	1.985
4	2.2	2.37
5	2.285	2.285
6	3.605	3.705

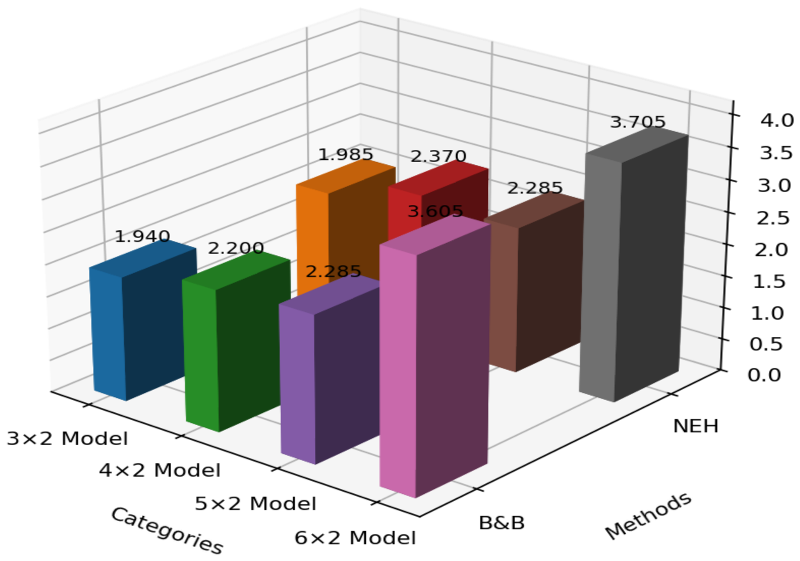

Fig. 2. Comparison Chart

The comparison shown in the figure clearly indicates that the proposed method performs better than the modified NEH heuristic. However, the relatively small difference in performance suggests that the modified NEH heuristic can still be a practical option when computational speed and scalability are key concerns.

IX. Conclusion

The proposed methodology ensures the best possible schedule for a flow shop. Yet, it is computationally intensive and becomes unfeasible for big issue sizes. The modified NEH heuristic delivers a near-optimal solution rapidly, sacrificing precision for efficiency and scalability in medium to large cases.

X. Future Work

Future research may explore the application of this result to hesitant fuzzy sets, interval-valued Pythagorean fuzzy sets, and interval-valued fermatean Fuzzy sets. Future work will involve a comparison between full fuzzy arithmetic and the current defuzzification-based approach by propagating interval-valued structures through scheduling computations.

Conflict of Interest:

There was no relevant conflict of interest regarding this paper.

References

I. Atanassov, Krassimir T. "Intuitionistic Fuzzy Sets." *Fuzzy Sets and Systems*, vol. 20, no. 1, 1986, pp. 87–96. 10.1016/S0165-0114(86)80034-3.

II. Atanassov, Krassimir, and George Gargov. "Interval Valued Intuitionistic Fuzzy Sets." *Fuzzy Sets and Systems*, vol. 31, no. 3, 1989, pp. 343-49. 10.1016/0165-0114(89)90205-4.

III. Dessouky, Maged M., et al. "Flowshop Scheduling with Identical Jobs and Uniform Parallel Machines." *European Journal of Operational Research*, vol. 109, no. 3, 1998, pp. 620-31. 10.1016/S0377-2217(97)00111-2.

IV. Dubois, Didier, and Henri Prade. "Systems of Linear Fuzzy Constraints." *Fuzzy Sets and Systems*, vol. 3, no. 1, 1980, pp. 37-48. 10.1016/0165-0114(80)90004-4.

V. Gholami-Zanjani, Seyed Mohammad, et al. "Robust and Fuzzy Optimisation Models for a Flow Shop Scheduling Problem with Sequence Dependent Setup Times: A Real Case Study on a PCB Assembly Company." *International Journal of Computer Integrated Manufacturing*, vol. 30, no. 6, 2017, pp. 552-63. 10.1080/0951192X.2016.1187293.

VI. Gupta, Deepak, and Sonia Goel. "Branch and Bound Technique for Two Stage Flow Shop Scheduling Model with Equipotential Machines at Every Stage." *International Journal of Operational Research*, vol. 44, no. 4, 2022, pp. 462-72. 10.1504/IJOR.2022.126588.

VII. Johnson, Selmer Martin. "Optimal Two- and Three-Stage Production Schedules with Setup Times Included." *Naval Research Logistics Quarterly*, vol. 1, no. 1, 1954, pp. 61–68. 10.1002/nav.3800010110.

VIII. Kurniawan, Latief Anggar, and F. Farizal. "Development of Flow Shop Scheduling Method to Minimize Makespan Based on Nawaz Enscore Ham (NEH) and Campbell Dudek and Smith (CDS) Method." *Proceedings of the 3rd African International Conference on Industrial Engineering and Operations Management*, 2022, pp. 1224-31. 10.46254/AF03.20220230.

IX. Lee, Gyu-Chang, and Yeong-Dae Kim. "A Branch-and-Bound Algorithm for a Two-Stage Hybrid Flowshop Scheduling Problem Minimizing Total Tardiness." *International Journal of Production Research*, vol. 42, no. 22, 2004, pp. 4731-43. 10.1080/0020754041233128584 1.

X. Linn, Richard, and Wei Zhang. "Hybrid Flow Shop Scheduling: A Survey." *Computers & Industrial Engineering*, vol. 37, no. 1-2, 1999, pp. 57-61. 10.1016/S0360-8352(99)00024-X.

XI. Malhotra, Khushboo, et al. "Bi-Objective Flow Shop Scheduling with Equipotential Parallel Machines." *Malaysian Journal of Mathematical Sciences*, vol. 16, no. 3, 2022, pp. 451-70. 10.47836/mjms.16.3.04.

XII. Nawaz, Muhammad, et al. "A Heuristic Algorithm for the M-Machine, N-Job Flow-Shop Sequencing Problem." *Omega*, vol. 11, no. 1, 1983, pp. 91-95. 10.1016/0305-0483(83)90088-9.

XIII. Ruiz, Rubén, et al. "Solving the Flowshop Scheduling Problem with Sequence Dependent Setup Times Using Advanced Metaheuristics." *European Journal of Operational Research*, vol. 165, no. 1, 2005, pp. 34-54. 10.1016/j.ejor.2004.01.041.

XIV. Ruiz, Rubén, and José Antonio Vázquez-Rodríguez. "The Hybrid Flow Shop Scheduling Problem." *European Journal of Operational Research*, vol. 205, no. 1, 2010, pp. 1-18. 10.1016/j.ejor.2009.09.024.

XV. Senapati, Tapan, et al. "Analysis of Interval-Valued Intuitionistic Fuzzy Aczel–Alsina Geometric Aggregation Operators and Their Application to Multiple Attribute Decision-Making." *Axioms*, vol. 11, no. 6, 2022, pp. 258. 10.3390/axioms11060258.

XVI. Xu, Ze Shui. "Methods for Aggregating Interval-Valued Intuitionistic Fuzzy Information and Their Application to Decision Making." Control and Decision, vol. 22, no. 2, 2007, pp. 215-19. https://www.researchgate.net/publication/271843255_Methods_for_aggregating_intervalvalued_intuitionistic_fuzzy_information_and_their_application_to_decision_making