

Modelling the Comparative Analysis of Johnson's Rule and Time Deviation Rule Heuristics with Exact Branch and Bound Approach Validation: A Case of Job Shop Sequencing Problems

Jairos Kahuru Shinzeh*

*Department of Mathematics and Statistics. The University of Dodoma, P.O. Box 338, Dodoma, Tanzania

E-mail of the corresponding author: jaiyus@gmail.com

Abstract

This study conducts a rigorous comparative analysis of two prominent flow shop scheduling heuristics the Johnson's Rule (JR) and the Time Deviation Rule (TDR) by validating their performance against an exact Branch and Bound (B&B) algorithm. The primary objective is to determine which heuristic provides solutions closest to the proven optimum for the key performance metrics of Total Elapsed Time (TET) and Total Idle Time (TIT). A computational experiment was designed using 15 benchmark problem instances of varying sizes and complexities. The exact B&B method was implemented to establish optimal solutions for each problem, serving as the validation benchmark. Both JR and TDR were applied to the same problem set. Performance was quantitatively evaluated by calculating the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for each heuristic's TET and TIT results against the B&B-derived optimum. Data analysis and visualization were performed using MATLAB. The B&B algorithm consistently generated superior schedules, confirming its value as a benchmark. The error analysis revealed a nuanced performance trade-off between the heuristics. For minimizing TET, JR (MAE = 2.27) slightly outperformed TDR (MAE = 2.40). However, for minimizing TIT, TDR demonstrated a significant advantage, with both lower MAE (5.27 vs. JR's 6.53) and lower RMSE (9.49 vs. JR's 11.25). The aggregate results indicate that TDR provides a more robust and accurate overall approximation for sequencing problems. This research provides a critical, empirically-grounded validation of classical scheduling heuristics against an exact optimization method. It moves beyond simple TET comparison to offer a multi-metric evaluation, demonstrating that the TDR is generally a more reliable heuristic than JR for comprehensive schedule quality, particularly in optimizing machine utilization through idle time reduction. To enhance the generalizability of these findings, future work should involve larger-scale and real-world problem datasets. Furthermore, investigating the scalability of these methods and integrating them with metaheuristics in particular Genetic Algorithms and Simulated Annealing could yield even more powerful and efficient hybrid scheduling solutions.

Keywords: Flow Shop Sequencing; Johnson's Rule; Time Deviation Rule; Branch and Bound Approach; Total Elapsed Time minimization; Heuristic Validation.

1. Introduction

Flow shop sequencing is a fundamental optimization problem in operational research and manufacturing logistics where a collection of jobs must be processed through a series of machines in a specific, fixed order (Jayasankari, 2021). This problem is a cornerstone for enhancing manufacturing efficiency, optimizing supply chain logistics, and improving overall resource allocation (Pinedo, 2016). Its practical applications are extensive, directly impacting the performance of real-world production systems where minimizing the total production time, or makespan, is a primary objective (Zhang et al., 2023). Key concepts in these sequencing problems include processing time, the time a job requires on a machine, Total Elapsed Time (TET) or makespan, the total time from the start of the first job to the end of the last job, and Total Idle Time (TIT), the time a machine remains free between jobs (Lakshmi, 2022; Watekar, 2017).

The resolution of flow shop scheduling problems is typically approached through two distinct classes of methods, the exact methods and heuristic methods. Exact methods, such as the Branch and Bound (B&B) algorithm, intelligently explore the solution space to guarantee an optimal solution but are often computationally prohibitive for large-scale problems due to the NP-hard nature of sequencing (AitZai, 2012). In contrast, heuristic methods provide computationally efficient, feasible solutions but do not guarantee optimality. Prominent among these heuristics for flow shops are Johnson's Rule (JR), which provides an optimal sequence under specific conditions (e.g., no job passing) (Garey et al., 1976), and the Time Deviation Rule (TDR) (Palmer, 1965). A critical research gap exists in the comprehensive validation of these heuristic rules against exact methodological benchmarks to rigorously quantify their performance and reliability in near-optimal sequencing.

The methodological approaches for this study are divided into heuristics and an exact method for validation. Heuristic Methods offer a nearly optimal solution in a fraction of the time required by exact methods. Johnson's Rule (JR) is a classic algorithm that generates an optimal sequence for two-machine flow shops where passing is not allowed (Lakshmi, 2022). Its application to larger machine sets or job shops requires modification to adopt the concept of fictitious machines, during determination of the optimal sequence. The Time Deviation Rule (TDR) is another constructive heuristic that prioritizes jobs based on the variance of their processing times across machines, operating on the rationale that jobs with highly variable times are more sensitive to their position in the sequence. The exact Branch and Bound (B&B) algorithm serves as the benchmark that

systematically enumerates potential solutions (branching) while using calculated bounds to prune sub-optimal paths, thereby guaranteeing to find the optimal solution without exhaustively evaluating every possibility (AitZai, 2012). Validating the results of JR and TDR against the B&B-derived optimum is essential for assessing their effectiveness.

The primary objectives of this paper are to model a specific flow shop sequencing problem with the goal of minimizing the Total Elapsed Time (TET). To apply and evaluate the performance of two prominent heuristics, the Johnson's Rule (JR) and the Time Deviation Rule (TDR) in solving this problem. To develop and implement an exact Branch and Bound (B&B) algorithm to establish the provably optimal solution for use as a validation benchmark. To conduct a rigorous comparative analysis of the heuristic methods against the B&B optimum using established performance metrics, namely Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), to quantify the deviation in TET and TTT and evaluate solution quality (Hyndman & Koehler, 2006; Hodson, 2022).

One of the main concerns in operations research is choosing efficient heuristics for flow shop scheduling, which is still a central concern. An examination of the empirical literature indicates a strong emphasis on the issues surrounding scalability, performance metrics, and heuristic reliability that have not yet been addressed. One of the main problems is that classical heuristics don't always work well in generalized "n-machine" settings. Johnson's Rule (JR) is proven to be the best option in the two-machine scenario, but in more complicated situations, its effectiveness decreases (Gupta and Stafford, 2006). Similar to this, the Time Deviation Rule (TDR) and other heuristics perform differently depending on the type of problem (Framinan et al., 2014). It is challenging to measure the true accuracy of heuristics because they are frequently compared against one another rather than a known optimum, which is a significant flaw shared by these studies. Secondly, the research shows that makespan minimization is the primary focus of the literature (Taillard, 1993), frequently at the expense of other important performance metrics. Improving machine utilization and energy efficiency requires reducing idle time just as much as makespan (Mouzon et al., 2007). Studies that do consider idle time, such as Teeravaraprug and Sakulpipat (2007), provide valuable insights into the trade-offs between methods like JR and Theory of Constraints (TOC) scheduling. However, without validation from an exact method like Branch and Bound (B&B), the conclusions about how close these heuristics get to the true optimal trade-off between makespan and idle time remain speculative. There is also a sizable scalability gap in the literature. Foundational comparative research (e.g., Nawaz et al., 1983; Kalczyński & Kamburowski, 2007) are frequently restricted to minor problem cases. This neglects the performance of heuristics on the larger, more complex problems characteristic of modern industrial applications, leaving their practical utility for real-world scenarios in question. It is clear from works such as Modrak et al. (2013) that examined a number of heuristics (e.g. NEH, Palmer, and CDS) but only examined makespan and lacked cross-validation against exact methods, providing an incomplete picture of their overall effectiveness.

A significant methodological gap is highlighted by the reviewed literature as a whole, namely the lack of a thorough, multi-metric validation of heuristic performance against an exact optimal solution for "n-machine" flow shops. This gap appears in three distinct ways, such as the validation gap, whereby prior research has not compared heuristic results (for idle time and makespan) to the actual optimum offered by an exact method such as Branch and Bound. Heuristic accuracy claims cannot be measured in the absence of this standard. Metric gap: Because idle time is a crucial metric for assessing scheduling effectiveness and resource usage, it has been overlooked as a result of the excessive focus on makespan minimization. The practical scalability of heuristics such as JR and TDR is uncertain due to the lack of research on how well they perform on larger, more industrially relevant problem sizes. The purpose of this study is to close this gap by performing a computational experiment that employs a B&B algorithm to determine the best solutions for benchmark problems. The performance of Johnson's Rule and the Time Deviation Rule is then thoroughly assessed and contrasted using makespan and idle time metrics. With this method, the most reliable technique for near-optimal flow shop scheduling will be found and a clear, quantitative assessment of each heuristic's dependability will follow.

This paper proceeds as follows: Section 2 explains the mathematical building process; Section 3 detailed methodology; Sections 4 present and discusses results and Section 5 is conclusion, recommendation and the direction of future research.

2. Problem Definition and Mathematical Formulations

This section defines a problem concept in regards to this work, develops a mathematical model with formal objective functions and constraints for the classic job-shop scheduling problem model and the exact Branch and Bound approach in the context of Job Shop Sequencing Problems and the performance measure metrics for model selection. Table 1 summarizes the mathematical notations and their descriptions used throughout the paper.

Table 1: Mathematical notations used in model formulation

Notation	Description
J	Set of jobs $J = (1, 2, 3, \dots, n)$ assigned by j
M	Set of machines $M = (1, 2, 3, \dots, m)$ assigned by i
n	Number of jobs to be processed
m	Number of machines in a shop
P_{ij}	Processing job (j) on machine i
C_{ij}	The completion time of job j
S_{ij}	The starting time of job j

2.1 Problem Definition

The job shop scheduling problem considered in this study involves a set of jobs $J = (1, 2, \dots, n)$ that must be processed on a set of machines $M = (1, 2, \dots, m)$. Each job j consists of a sequence of operations, $O_{j1}, O_{j2}, \dots, O_{jm}$ that must be processed in a specified order, with each operation O_{ij} requiring a specific processing time p_{ij} on machine i . The primary objective is to minimize the TET, defined as the maximum completion time of all jobs. The problem is subject to the following constraints: Machine constraint (each machine can process at most one operation at a time), operation constraint (each operation must be completed without interruption once started) and precedence constraint (for each job, operations must be processed in the specified sequence).

2.2 Mathematical Formulations

This section develops a mathematical model with formal objective function and constraints for exact Branch and Bound approach in the context of Job Shop Sequencing Problems (JSSP). Branch and Bound (B&B) method provides an exact solution approach for JSSP by systematically enumerating potential solutions while pruning branches that cannot yield better solutions than those already found (Brucker et al., 1997; Guzmán-Ortiz, Andres & Poler, 2022). The methodology by Abdolrazzagh-Nezhad & Abdullah (2017) and Pei et al (2022) was considered as a supplementary knowledge in formulating the model below:

Decision Variables:

- S_{ij} : Start time of job j on machine i
- $C_{ij} = S_{ij} + P_{ij}$: Completion time of job j on machine i

Objective function:

- Minimize the time to process all jobs on all machines for a given sequence of jobs.

$$\text{Minimize}(TET) = \text{Minimize}\left(\max_j C_{mj}\right) \quad (1)$$

Subject to

- Precedence constraints (job operation sequence):

$$S_{ij} + P_{ij} \leq S_{(i+1)j} \quad \forall j, \forall i \in \{1, \dots, m-1\} \quad (2)$$

- Machine capacity constraints (no overlap on the same machine):

$$S_{ij} + P_{ij} \leq S_{ik} \text{ or } S_{ik} + P_{ik} \leq S_{ij} \quad \forall j \neq k, \forall i \quad (3)$$

- Lower bound (used to prune branches):

$$LB = \max \left\{ \max_i \left(\sum_j P_{ji} \right), \max_j \left(\sum_i P_{ji} \right) \right\} \quad (4)$$

- Incumbent solution constraints (Branch and Bound Pruning)

$$LB \geq \text{Current best } TET \quad (5)$$

- Non-negativity and feasibility constraints

$$S_{ij} \geq 0, \quad \forall i, j \quad (6)$$

2.3 Mean Absolute Error

Mean Absolute Error (MAE) the average of the absolute differences between predicted values and actual values that measures the average magnitude of errors in a set of predictions, without considering their direction. In this context, it measures the average absolute difference between predicted heuristic and actual B&B benchmark values. The metric is easy to interpret and robust to outliers (Willmott & Matsuura, 2005). Consider the following mathematical formula:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (7)$$

where: n : is the number of data points

y_i : is the actual value

$\hat{y}_i$: is the predicted value

2.4 Root Mean Squared Error

Root Mean Squared Error (RMSE) is the square root of the Mean Squared Error (MSE). It brings the metric back to the original scale of the data, making it easier to interpret than MSE. The square root of MSE, providing error magnitude in the original units (Hodson, 2022). RMSE is more interpretable than MSE and widely used in model evaluation (Hyndman & Koehler, 2006). Consider its mathematical formula below:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (8)$$

where: n : is the number of data points

y_i : is the actual value

$\hat{y}_i$: is the predicted value

3. Methodology

The methodology consists mainly four phases: (1) Algorithm description and adaptation, (2) Case study and data collection (3) Performance metric and heuristic evaluation (4) Heuristic selection and decision rule. In evaluating the performance of the JR and TDR, the consideration is on a sequencing problem with **n-jobs** to be processed on three machines (M_1, M_2 and M_3) or **m-machines** ($M_1, M_2, \dots, M_m$) with each job having a known processing time on all the machines.

3.1 Johnson's Rule Algorithm

Johnson (1954) studied a two machine **n-job** flow shop problem and gave an optimal algorithm for a sequence giving minimum completion time for all jobs. He subsequently extended the optimal algorithm for a special type of three machine and the general **n-job** to **m-machine** flow shop problem. The algorithm under Johnson's rule was adopted from Okwu and Emoyon (2020), Rahmana et al. (2023) and Babatunde & Onikoyi (2016). Processing **n-job** on **3-machine** and **n-job** on **m-machine** algorithms are almost similar because both involves optimality testing and the generation of fictions machines during optimal job sequence determination (Gupta et al., 2011; Asif et al., 2022). Here we present the algorithm involving processing **n-job** through **m-machine**. The following are the relevant steps:

Step 1: Testing for optimality. The optimal solution can be obtained if either or both of the following conditions hold good. That is, the minimum processing time on machines M_1 and machine M_m is as greater as the maximum processing time on any of the remaining $(m-1)$ machines. Find;

- $\text{Min}(t_{1j}), \text{Min}(t_{mj})$ and $\text{Max}(t_{ij})$ to verify the below conditions such that;
- $\text{Min}(t_{ij}) \geq \text{Max}(t_{ij}); j = 2, 3, \dots, (m-1)$ and/or
- $\text{Min}(t_{mj}) \geq \text{Max}(t_{ij}); j = 2, 3, \dots, (m-1)$.

Step 2: If either or both the conditions mentioned above hold, then go to step 3. Otherwise, the algorithm fails.

Step 3: Convert **m-machine** problem into **2-machine** problem by introducing two fictitious machines (G and H) with corresponding processing times given by:

- $tG_j = t_{1j} + t_{2j} + \dots + t_{m-1j}; j = 1, 2, \dots, n$, i.e. processing time of n-jobs on machine G is the sum of the processing times on machines $M_1, M_2, \dots, M_{m-1}$
- $tH_j = t_{2j} + t_{3j} + \dots + t_{mj}; j = 1, 2, \dots, n$, i.e. processing time of n-jobs on machine H is the sum of the processing times on machines $M_1, M_2, \dots, M_{m-1}$

Step 4: The same procedure for solving **2-machine** problem can be used to find the optimal sequence for the fictions machines G and H .

Step 5: Select the job having the shortest activity time. If the activity selected lies under first work center, schedule the job first and if the activity selected lies under the second work center, schedule the job last.

Step 6: The selected job must be eliminated from the list as per step 5. Repeat step 5, till all selected jobs are scheduled.

3.2 Time Deviation Rule Algorithm

Time deviation method is used to obtain the optimal sequence of the jobs. In this method time duration table is calculated for each job in the row wise and the column wise (Rao, Raju & Babu (2013). Karthikeyan (2010), Watekar (2017) and Mosheiov (2004) in their respective papers discussed the Time Deviation Rule (TDR) for processing n – job through 2 – machine , n – job through 3 – machine and n – job through m – machine in determining the optimal sequence of jobs in job sequencing problems. The following are the steps in implementing the TDR method:

Step 1: Determine row and column maximum times for each cell (i, j) of the problem matrix.

$$p_i = \max_{j=1}^m t_{ij}, c_j = \max_{i=1}^n t_{ij} \quad (9)$$

Step 2: Determine row and column deviation for each cell (i, j) of the problem matrix.

$$r_{ij} = p_i - t_{ij} \quad (10)$$

$$s_{ij} = c_j - t_{ij} \quad (11)$$

Where;

r_{ij} is the row time deviation of $(i, j)^{th}$ cell

p_i is the maximum time of i^{th} row

s_{ij} is the column time deviation of $(i, j)^{th}$ cell

c_j is the maximum time of j^{th} column

t_{ij} is the time required to process i^{th} job on j^{th} machine

Step 2: Find the cell with both time deviation vectors are zero for machine M_1 and perform the corresponding job firstly.

Step 3: Suppose more than one cell has both deviation vectors are zero then obtain sum deviations of the corresponding columns. Perform the job first which cell have the largest sum deviation and perform next job which have next largest sum deviation and so on.

Step 4: Similarly look the cell which has time deviation vectors both are zero for machine M_2 and perform the corresponding job lastly.

Step 5: If more than one cell has both zero deviations, then obtain sum deviations of the corresponding columns. Perform the job last with cell that have the largest sum deviation and perform next job which have next largest sum deviation prior to last and so on.

Step 6: If we get the sequence involving, all jobs for the job sequencing problem, then stop the process. Otherwise, go to next step.

Step 7: Form the reduced time duration table which contains only non-assigned jobs.

Step 8: Continue with the above steps from (1) to (6) for the reduced time duration table. But perform the jobs next to the previously assigned jobs for the cells which have both time deviation vectors are zero for M_1 machine and perform jobs previous to the last assigned jobs for cells which have both time deviation vectors are zero for M_2 machine.

Step 9: This will be continued until we get the required order of sequence of jobs for job sequencing problems, which will give minimum total elapsed time.

3.3 Exact Branch and Bound Algorithm

This algorithm is a well-known exact algorithm for solving an optimization problem, especially combinatorial optimization that obtains the optimal solution by dividing the original problem into a series of sub problems using certain criteria (Takano & Nagano, 2017; Daneshamooz et al., 2021; Shackelton, 1971). Shackelton (1971), Gupta and Goel (2022) and Kouider & Haddadène (2021) outlined the series of steps to find the optimal solution of the problem.

Step 1: Initialization: Determine the initial Upper Bound (UB), create root node with all jobs unscheduled and initialize active nodes list with root node.

Step 2: Node Selection: Select node with smallest UB (best-first search)

Step 3: Branching: For selected node, create child nodes by fixing one more job in sequence for level k , create $n - k$ child nodes for remaining unscheduled jobs.

Step 4: Bounding: For each child node: Calculate LB, If $LB \geq UB$, prune this Branch. If complete sequence and $TET \geq UB$, update upper bound.

Step 5: Termination: When no active nodes remain, the current UB value is the optimal solution.

3.4 Case study Description and Data collection

Fifteen (15) benchmark problem instances were selected from the literature on flow shop scheduling that was available. These instances were chosen to represent a diverse set of conditions, including varying complexity levels, balanced and unbalanced processing time matrices, and varying job counts (n), in order to ensure a robust evaluation of the algorithms.

3.5 Performance Metrics and Heuristic Evaluation

Following the application of Johnson's Rule (JR) and the Time Deviation Rule (TDR) algorithms, each heuristic yields an optimal job sequence. This sequence serves as the input for calculating the key performance metrics: Total Elapsed Time (TET) and Total Idle Time (TIT). The calculation of TET and TIT for a given sequence was performed using a computational matrix algorithm. This method provides a systematic, tabular approach for determining the start and finish times of all operations on all machines, from which TET and TIT are directly derived. For this study, the algorithms were coded in MATLAB to compute the TET and TIT for JR, TDR, and the exact Branch and Bound (B&B) method and the results for all problem instances are presented in Table 2. To quantify the performance of the heuristics, the B&B results were established as the optimal benchmark. The absolute deviations in TET and TIT for the JR and TDR solutions were calculated against this benchmark for each problem instance, as detailed in Table 3. These deviation values were subsequently used to compute the aggregate error metrics Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) providing a robust measure of each heuristic's average performance and error magnitude, respectively (see Table 4).

3.6 Heuristic Selection and Decision Rule

The selection between Johnson's Rule (JR) and the Time Deviation Rule (TDR) was based on a quantitative analysis of their performance deviations from the proven optimal solutions generated by the Branch and Bound (B&B) algorithm. The Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) for both Total Elapsed Time (TET) and Total Idle Time (TIT) served as the key comparative metrics. The formal decision rule for method selection is as follows:

- **Primary Criterion:** The heuristic with the lowest MAE and RMSE values is preferred, as this indicates the smallest average deviation from the optimal benchmark across all problem instances (Hodson, 2022; Chai & Draxler, 2014).
- **Unambiguous Case:** If one heuristic demonstrates superior performance (lower errors) for both TET and TIT, it is conclusively selected as the preferred method (Greco, Matarazzo & Slowinski, 2000).
- **Conflicting Case:** If the results are conflicting (e.g., one heuristic is better for TET while the other is better for TIT), priority is given to the heuristic that minimizes TET. This prioritization is justified because minimizing (TET) is typically the primary objective in flow shop scheduling, as it directly correlates with throughput and delivery times (Lee, 1997; Kubiak et al., 2002).

4. Results and Discussion

Table 2 provides TET and TIT solutions for three methods Johnson's Rule (JR), Time Deviation Rule (TDR) and the Exact Branch and Bound (B&B) approach in fifteen (15) conveniently selected sequencing problems.

Table 2: Metric solution for JR, TDR and B&B methods

Example	Method/Heuristic	TET	TIT
1.	Johnson's rule	57	53
	Time deviation rule	51	44
	Exact Branch and Bound	51	46
2.	Johnson's rule	121	281
	Time deviation rule	116	280
	Exact Branch and Bound	112	246
3.	Johnson's rule	55	83
	Time deviation rule	55	75
	Exact Branch and Bound	51	79
4.	Johnson's rule	87	87
	Time deviation rule	85	81
	Exact Branch and Bound	84	81
5.	Johnson's rule	45	19
	Time deviation rule	45	19
	Exact Branch and Bound	43	16
6.	Johnson's rule	145	28
	Time deviation rule	144	31

	Exact Branch and Bound	142	29
7.	Johnson's rule	45	12
	Time deviation rule	48	9
	Exact Branch and Bound	44	9
8.	Johnson's rule	66	37
	Time deviation rule	65	23
	Exact Branch and Bound	62	14
9.	Johnson's rule	44	10
	Time deviation rule	50	5
	Exact Branch and Bound	44	10
10.	Johnson's rule	72	42
	Time deviation rule	74	35
	Exact Branch and Bound	72	39
11.	Johnson's rule	59	22
	Time deviation rule	64	17
	Exact Branch and Bound	59	22
12.	Johnson's rule	50	41
	Time deviation rule	50	41
	Exact Branch and Bound	50	39
13.	Johnson's rule	84	81
	Time deviation rule	85	81
	Exact Branch and Bound	84	81
14.	Johnson's rule	96	46
	Time deviation rule	96	54
	Exact Branch and Bound	96	54
15.	Johnson's rule	53	19
	Time deviation rule	53	19
	Exact Branch and Bound	51	16

Figure 1 and Figure 2, compares the performances of TET and TTT for the three methods B&B, JR and TDR to identify or verify the benchmark method.

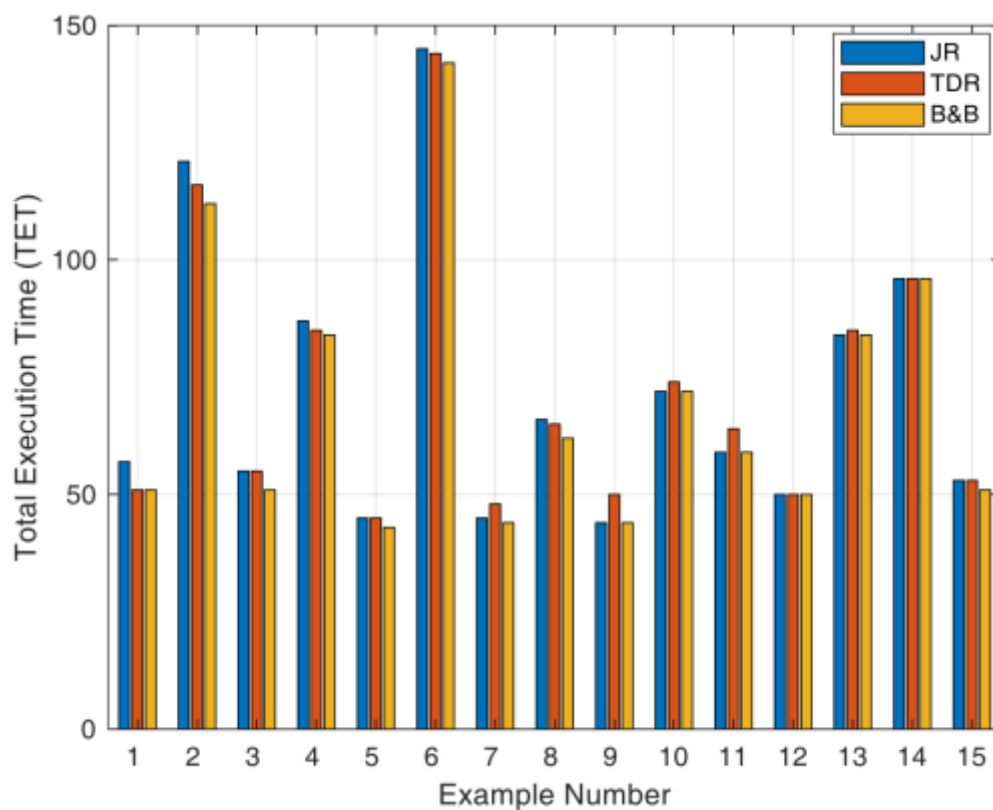


Figure 1: Comparison on TET for JR, TDR and B&B methods

Figure 1 provides the bar chart comparing the Total Elapsed Time (TET) of three scheduling methods; JR, TDR), and B&B. The results shows that, B&B consistently achieves the lowest TET in most examples such as 2, 3, 5, 7, 8, 9, 11 and 15, indicating it is the most superior to others.

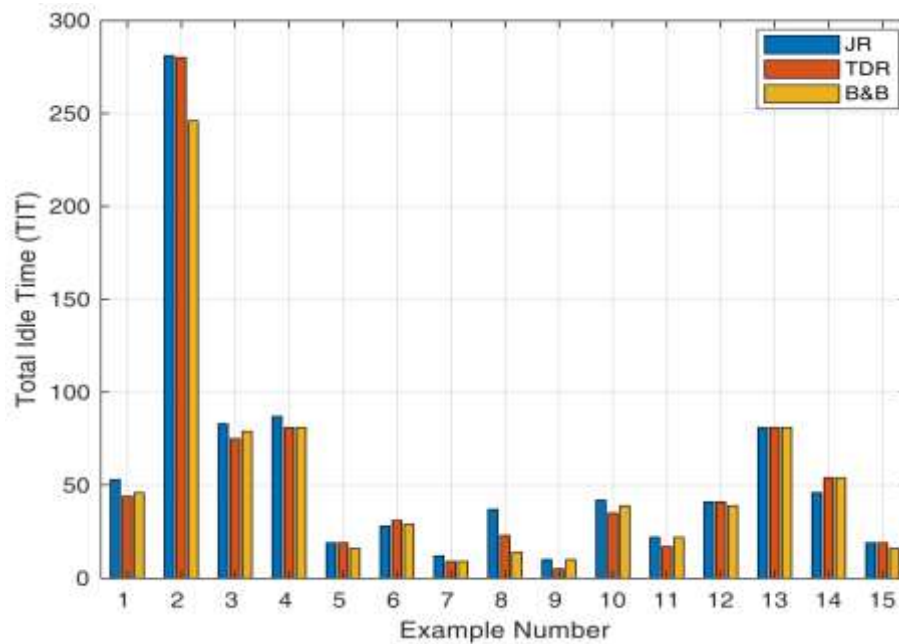


Figure 2: Comparison on TIT for JR, TDR and B&B methods

Figure 2 provides a bar chart comparing Total Idle Time (TIT) for the three scheduling methods JR, TDR and B&B. According to examples 2, 5, 8, 12 and 15 B&B is expected to outperform other heuristics JR and TDR as it has the lowest values of TIT, which is a criterion for best performance.

Table 3: Percentage Score on wins and ties of TET and TIT for JR, TDR and B&B

Method (Heuristic)	TET		TIT		Overall, Score (%)
	Wins (out of 15)	Score (%)	Wins (out of 15)	Score (%)	
B&B	10	66.7	11	73.3	70
TDR	4	26.7	1	6.7	16.7
JR	1	6.6	3	20	13.3

Table 3 displays a combined summary for percentage comparison scores of JR, TDR, and B&B methods based on TET and TIT. The overall percentage scores shows that, B&B is the most effective method against JR and TDR, dominating both TET and TIT performances. Therefore, B&B is superior aligning with its exact optimization nature.

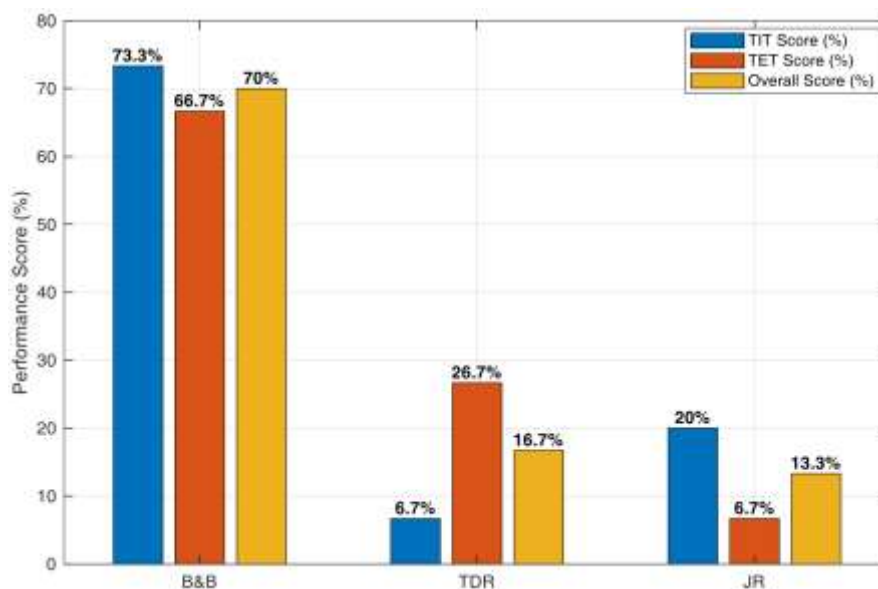


Figure 3: Comparison of Scheduling Methods: TIT, TET, and Overall Scores

Figure 3 compares performances in percentage score for B&B, TDR, and JR methods across three different score types, TIT score (in blue), TET score (in orange) and Overall score (in yellow). The comparison is based on wins (number of minimum TET and TIT values, plus number of ties on TET and TIT values on both categories). B&B performs better in TIT (73.3%)

but slightly better, in TET (66.6%), TDR performs better in TET (26.7%) but poorly in TIT (6.7%). JR has better performance in TIT (20%) than in TET (6.7%). The overall performance indicate that B&B scores (70%), TDR scores (16.7%) and JR scores (13.3%), which signifies that B&B is the top performer in all metrics (TET and TIT), qualifying to be superior aligning with its exact optimization nature. Therefore, we reserve the B&B (TET and TIT) solution for validation purpose in order to make JR and TDR methods to compete on their performances.

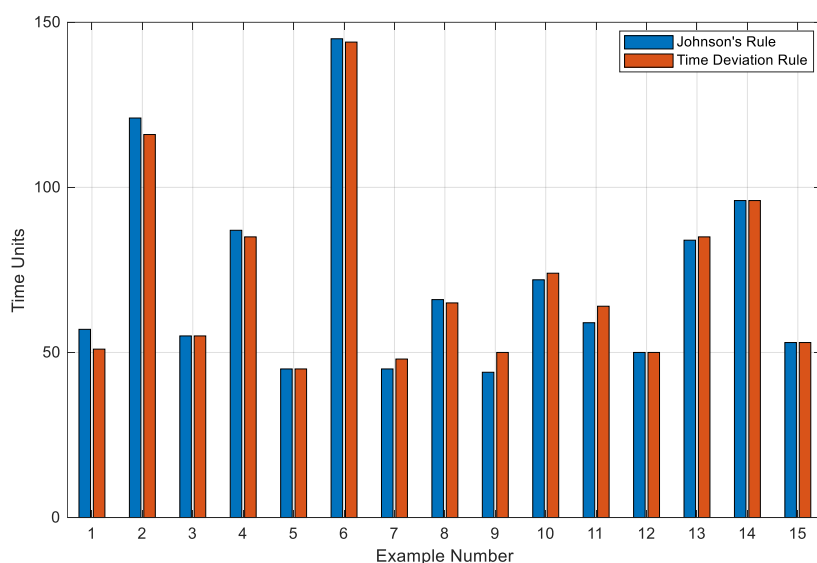


Figure 4: Comparison on TET solution for Johnson's Rule and Time Deviation Rule

The bar graph in Figure 4 compares the TET performance of two scheduling heuristics the JR and TDR. The preliminary investigation shows that, TDR outperforms JR, by achieving a lower TET in 7 scores out of 15 as reflected in examples 1, 2, 4, 6, 8, 12 and 13. JR ties TDR in examples 3, 5, 14 and 15, signifying no heuristic benefit in these cases. TDR underperforms JR in examples 7, 9, 10 and 11, indicating situational limitations. TDR is generally superior to JR for minimizing TET, particularly in problems like 1, 2 and 8. JR remains competitive in cases like in 5 and 15.

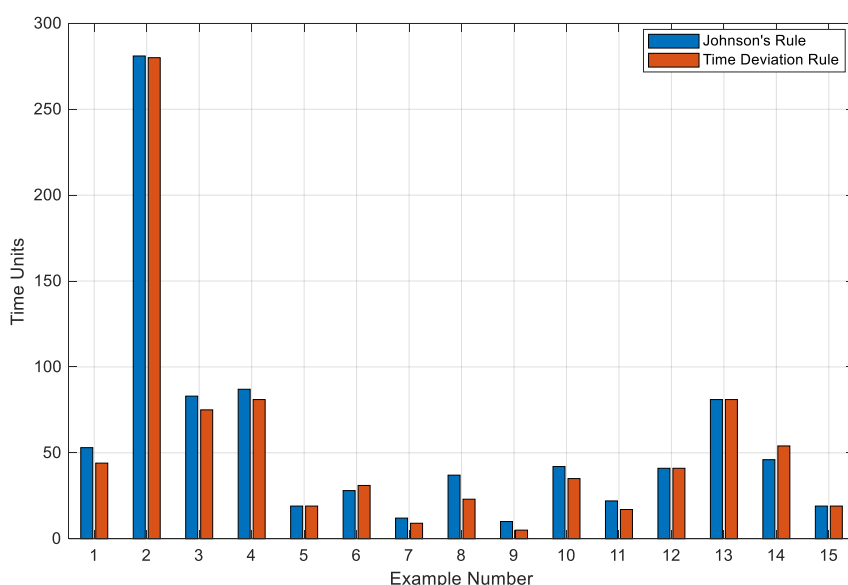


Figure 5: Comparison on TIT solution for Johnson's Rule and Time Deviation Rule

The bar graph in Figure 5 compares the performance of two scheduling methods JR and TDR using TIT solution. TDR reduces TIT more efficiently than JR in most cases, outperforming JR in 9 out of 15 cases as shown in examples 1, 3, 4, 7, 8, 9, 10, 11 and 12. JR matches TDR in examples 2, 5, 13 and 15, showing no heuristic benefit. JR is more stable than TDR in cases 6 and 14.

Table 4: Deviations for TET (JR & TDR) and TIT (JR & TDR)

Example	Optimal TET (B&B)	JR Error	TDR Error	Optimal TIT (B&B)	JR Error	TDR Error
1	51	6	0	46	7	-2

2	112	9	4	246	35	34
3	51	4	4	79	4	-4
4	84	3	1	81	6	0
5	43	2	2	16	3	3
6	142	3	2	29	-1	2
7	44	1	4	9	3	0
8	62	4	3	14	23	9
9	44	0	6	10	0	-5
10	72	0	2	39	3	-4
11	59	0	5	22	0	-5
12	50	0	0	39	2	2
13	84	0	1	81	0	0
14	96	0	0	54	-8	0
15	51	2	2	16	3	3

Table 4 systematically show the calculations of TET and TIT deviations from B&B optimized schedules as a point of reference. Optimal TET (B&B) is referred from the exact B&B (TET) column. JR Error (TET) is obtained by, JR (TET) minus Optimal (TET). TDR Error (TET) is obtained by, TDR (TET) minus Optimal (TET). Optimal TIT (B&B) is referred from the exact B&B (TIT) column. JR Error (TIT) is obtained by taking JR (TIT) minus Optimal (TIT) and TDR Error (TIT) is obtained by taking TDR (TIT) minus Optimal (TIT). The negative error values at some point show that, some heuristic methods yielded a better value than the optimal B&B value.

Table 5: MAE and RMSE values on TET and TIT

Metric	JR	TDR
MAE (TET)	2.27	2.40
RMSE (TET)	3.43	3.01
MAE (TIT)	6.53	5.27
RMSE (TIT)	11.25	9.49

Table 5 shows the metrics MAE and RMSE for JR and TDR validated by the exact B&B optimal solution for both TET and TIT whereby, the MAE and RMSE with lower values indicates the better performance. For TET dataset, JR has MAE (2.27) and TDR MAE (2.40), implying that, JR has better performance. JR has RMSE (3.43) and TDR has RMSE (3.01) implying that, TDR has better performance. For the TIT dataset, JR has MAE (6.53) and TDR has MAE (5.27) showing that, TDR is significantly better. JR has RMSE (11.25) and TDR has RMSE (9.49), TDR is better. Generally, using MAE on TET, the performance is conflicting in the sense that, JR slightly outperforms TDR, but using RMSE, TDR performs better. On TIT, TDR constantly outperforms JR by a clear margin. In this case, Time Deviation Rule (TDR) appears to be inconclusively better heuristic than its counterpart Johnson's Rule (JR).

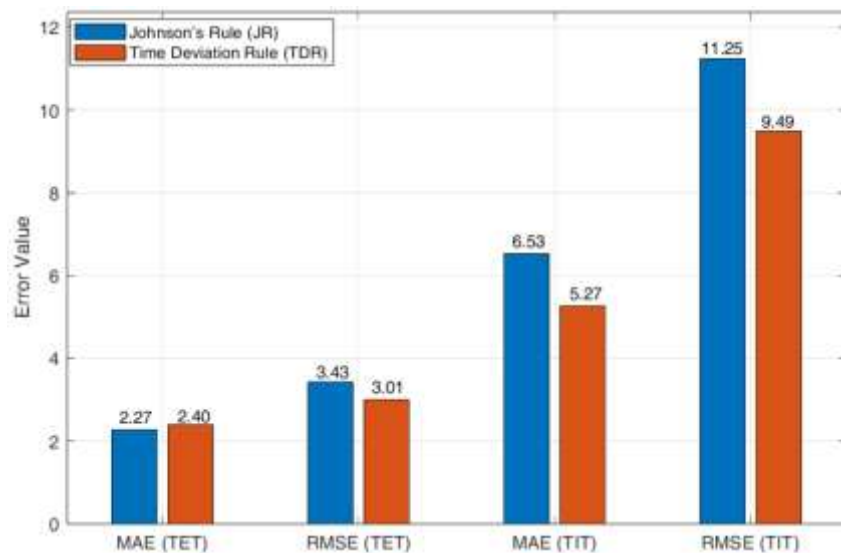


Figure 6: Performance comparison on JR and TDR using MAE and RMSE

Figure 6 shows the conflicting results on TET that might affect decision-making process in arriving to a good conclusion “using MAE, JR slightly outperforms TDR, but using RMSE, TDR outperforms JR”. To address this for clarity, we create three bar charts that average error values and provide a clearer comparison between JR and TDR by calculating average TET error, TIT error and overall error for each method and display three bar charts, one for TET errors, second for TIT errors and the third chart for overall average error. Figure 7 shows a clear visualization for JR and TDR on TET, TIT and overall errors.

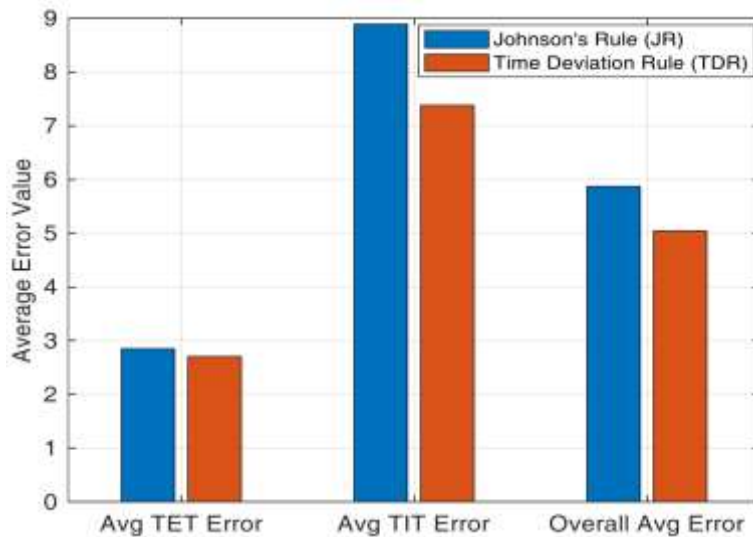


Figure 7: Average error comparison between JR and TDR heuristics

Figure 7 is a bar chart that presents a comparison of average error values between Johnson's Rule (JR) and the Time Deviation Rule (TDR) across three metrics, average error (TET), average error (TIT) and average error combined (overall error). For all the three metrics, average TET error (JR) is higher than average TET Error (TDR). TDR performs slightly better in average TET error and clearly better in average TIT error and has the lowest overall average error, indicating better general performance. In this case, clearly TDR is superior to JR across all averaged error metrics, principally in reducing TIT and the overall average error.

5. Conclusion and Recommendation

This study conducted a rigorous, multi-metric validation of two classical scheduling heuristics Johnson's Rule (JR) and the Time Deviation Rule (TDR) against the exact benchmark provided by a Branch and Bound (B&B) algorithm. The objective was to determine which heuristic provides solutions closest to the proven optimum for both Total Elapsed Time (TET) and Total Idle Time (TIT) across a diverse set of 15 flow shop problems. The B&B algorithm consistently generated superior schedules, confirming its effectiveness as a validation benchmark against the inherent sub-optimality of heuristic approaches. For the primary objective of minimizing TET, Johnson's Rule demonstrated a marginal advantage, as indicated by its slightly lower Mean Absolute Error (MAE = 2.27 vs. TDR's 2.40). However, for the critical secondary objective of minimizing TIT a key indicator of machine utilization and operational efficiency the TDR proved significantly more robust, outperforming JR on both MAE (5.27 vs. 6.53) and RMSE (9.49 vs. 11.25). When performance is aggregated across both metrics, TDR emerges as the more reliable and comprehensive heuristic. TDR is recommended for general application in flow shop environments, its strength in minimizing idle time directly translates to improved machine utilization and reduced operational costs, making it a more versatile tool than JR for holistic schedule optimization. Moreover, for any scenario requiring a balance between throughput (TET) and efficiency (TIT), TDR should be the heuristic of choice. For this paper, future studies should consider expanding the scale of analysis by employing larger and more complex benchmark problem sets (e.g., with 20+ jobs and more machines) to thoroughly test the scalability and robustness of both heuristics under more industrially relevant conditions. Incorporate real-world industrial data set to validate these findings that include stochastic elements such as machine breakdowns, dynamic job arrivals, and sequence-dependent setup times. Develop hybrid metaheuristic models to investigate the integration of these heuristics as initial seeds (e.g., Genetic Algorithms, Simulated Annealing). Formalize the trade-off between TET and TIT into a multi-objective optimization framework, using techniques like Pareto front analysis to provide schedulers with a set of optimal solutions representing different balances between the two competing objectives.

References

1. Abdolrazzagh-Nezhad, M., & Abdullah, S. (2017). Job shop scheduling: Classification, constraints and objective functions. *International Journal of Computer and Information Engineering*, 11(4), 429-434.
2. AitZai, A., Benmedjdoub, B., & Boudhar, M. (2012). A branch and bound and parallel genetic algorithm for the job shop scheduling problem with blocking. *International Journal of Operational Research*, 14(3), 343-365.
3. Asif, M. K. A., Alam, S. T., Jahan, S., & Arefin, M. R. (2022). An empirical analysis of exact algorithms for solving non-preemptive flow shop scheduling problem. *International journal of research in industrial engineering*, 11(3), 306-321.
4. Babatunde, B. O., & Onikoyi, I. A. (2016). Job Sequencing Methods And Total Elapsed Time Management In Block

- Production Industry. *STRATEGII MANAGERIALE*, 28.
5. Baker, K. R., & Trietsch, D. (2018). *Principles of sequencing and scheduling*. John Wiley & Sons.
6. Brucker, P., Hurink, J., Jurisch, B., & Wöstmann, B. (1997). A branch & bound algorithm for the open-shop problem. *Discrete Applied Mathematics*, 76(1-3), 43-59.
7. Bülbül, K., & Kaminsky, P. (2013). A linear programming-based method for job shop scheduling. *Journal of Scheduling*, 16, 161-183.
8. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247-1250.
9. Daneshmooz, F., Fattahi, P., & Hosseini, S. M. H. (2021). Mathematical modeling and two efficient branch and bound algorithms for job shop scheduling problem followed by an assembly stage. *Kybernetes*, 50(12), 3222-3245.
10. Framinan, J. M., Leisten, R., & Ruiz-Usano, R. (2014). *Flow Shop Scheduling: Theoretical Results, Algorithms, and Applications*. Springer.
11. Gangwar, A., & Kumar, A. (2018) Solved M-Machines, N-Job Flow Shop Sequencing Problem Using A Heuristic Algorithm.
12. Garey, M. R., Johnson, D. S., & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, 1(2), 117-129.
13. Greco, S., Matarazzo, B., & Slowinski, R. (2000). Decision rules. In *Encyclopedia of Management* (4th ed., pp. 178-183). Gale Group.
14. Gupta, D., & Goel, S. (2022). Branch and bound technique for two stage flow shop scheduling model with equipotential machines at every stage. *International Journal of Operational Research*, 44(4), 462-472.
15. Gupta, J. N. D., & Stafford, E. F. (2006). Flowshop scheduling research after five decades. *European Journal of Operational Research*, 169(3), 699-711.
16. Gupta, D., Sharma, S., & Sharma, S. (2011). Heuristic approach for n-jobs, 3-machines flow shop scheduling problem, processing time associated with probabilities involving transportation time, break-down interval, weightage of jobs and job block criteria. *Mathematical Theory and Modeling*, 1(1), 30-36.
17. Guzmán-Ortiz, B. E., Andres, B., & Poler, R. (2022). Matheuristic Algorithm for Job-Shop Scheduling Problem Using a Disjunctive Mathematical Model. *Computers*, 11(1).
18. Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development*, 15(14), 5481-5487.
19. Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679-688.
20. Jayasankari, S. (2021). An efficient flow shop scheduling problem with makespan objective. *Turkish journal of computer and mathematics education (TURCOMAT)*, 12(4), 461-466.
21. Ji, B., Zhang, S., Yu, S. S., & Zhang, B. (2023). Mathematical modeling and a novel heuristic method for flexible job-shop batch scheduling problem with incompatible jobs. *Sustainability*, 15(3), 1954.
22. Johnson, S. M. (1954). Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1(1), 61-68.
23. Kalczynski, P. J., & Kamburowski, J. (2007). On the NEH heuristic for minimizing the makespan in permutation flowshops. *Omega*, 35(1), 53-60.
24. Karthikeyan, K. (2010). Heuristics algorithms for job sequencing problems. *Global J. Sci. Front. Res. GJSFR Classif.-D*, 10, 76-86.
25. Khatami, M., Salehipour, A., & Hwang, F. J. (2019). Makespan minimization for the m-machine ordered flow shop scheduling problem. *Computers & Operations Research*, 111, 400-414.
26. Kouider, A., & Haddadène, H. A. (2021). A bi-objective branch-and-bound algorithm for the unit-time job shop scheduling: a mixed graph coloring approach. *Computers & operations research*, 132, 105319.
27. Kubiak, W., et al. (2002). The two-machine flowshop problem with limited machine availability: A branch and bound algorithm. *European Journal of Operational Research*, 137(3), 535-552.
28. Lakshmi, V. T. (2022). Finding Minimum total elapsed Time in a Sequencing Problem by a New Zero Identifier Method. *International Journal of Novel Research and Development*, 7.
29. Lee, C.-Y. (1997). Minimizing the makespan in the two-machine flowshop scheduling problem with an availability constraint. *Operations Research Letters*, 20(3), 129-139.
30. Maassen, K., Perez-Gonzalez, P., & Günther, L. C. (2020). Relationship between common objective functions, idle time and waiting time in permutation flow shop scheduling. *Computers & Operations Research*, 121, 104965.
31. Modrak, V., Semanco, P., & Kulpa, W. (2013). Performance measurement of selected heuristic algorithms for solving scheduling problems. *2013 IEEE 11th international symposium on applied machine intelligence and informatics (SAMI)* (pp. 205-209). IEEE.
32. Mosheiov, G. (2004). Simultaneous minimization of total completion time and total deviation of job completion times. *European Journal of Operational Research*, 157(2), 296-306.
33. Mouzon, G., Yildirim, M. B., & Twomey, J. (2007). Operational methods for minimization of energy consumption in manufacturing systems. *International Journal of Production Research*, 45(18-19), 4247-4271.
34. Nawaz, M., Ensco, E. E., & Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, 11(1), 91-95.
35. Nearchou, A. C. (2004). Flow-shop sequencing using hybrid simulated annealing. *Journal of Intelligent Manufacturing*, 15(3),

- 317–328.
36. Okwu, M., & Emovon, I. (2020). Application of Johnson's algorithm in processing jobs through two-machine system. *Journal of Mechanical and Energy Engineering*, 4(1), 33-38
 37. Palmer, D. S. (1965). Sequencing jobs through a multi-stage process in the minimum total time. *Operational Research Quarterly*, 16(1), 101–107.
 38. Pei, F., Zhang, J., Mei, S., & Song, H. (2022). Critical review on the objective function of flexible job shop scheduling. *Mathematical Problems in Engineering*, 2022(1), 8147581.
 39. Pinedo, M. L. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer.
 40. Rahmana, M., Murada, M. M. A., Rahmana, S., Roya, R., Muhammada, S., & Rahmanb, S. A. (2023). Application of Johnson's Algorithm for Scheduling Optimization and Finding an Optimal Sequence: A Case Study on a Shoe Factory.
 41. Rao, N. N., Raju, O. N., & Babu, I. R. (2013). Modified heuristic time deviation technique for job sequencing and Computation of minimum total elapsed time. *International journal of computer science & information technology*, 5(3), 67.
 42. Rocha, P. L., Ravetti, M. G., Mateus, G. R., & Pardalos, P. M. (2008). Exact algorithms for a scheduling problem with unrelated parallel machines and sequence and machine-dependent setup times. *Computers & Operations Research*, 35(4), 1250–1264.
 43. Shackelton, N. J. (1971). *Approaches to the n-job m-machine scheduling problem* (Doctoral dissertation, Monterey, California; Naval Postgraduate School).
 44. Taillard, E. (1993). Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2), 278–285.
 45. Takano, M. I., & Nagano, M. S. (2017). A branch-and-bound method to minimize the makespan in a permutation flow shop with blocking and setup times. *Cogent engineering*, 4(1), 1389638.
 46. Teeravaraprug, J., & Sakulpipat, T. (2007). A Comparison of Johnson's Rule and TOC Based Scheduling Methodologies.
 47. Watekar, A. B. (2017). Job sequencing. *Research Journal of Mathematical and Stat*, 5(5), 9-13.
 48. Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82.
 49. Zhang, T., Rose, O., & Pickl, S. (2023). Material flow control on shop floors in job shops: Release, routing, and sequencing. *Journal of Manufacturing Systems*, 66, 1–15.