A Model for Evaluating the Impact of Priority Rules on Flow Time and Wait Time In A Job Shop Scheduling System: A Single Machine Case

Muhammad Usman Nisar[1], Andi Cakravastia Arisaputra Raja[1], Anas Ma'ruf[1], Abdul Hakim Halim[1]

[1] Faculty of Industrial Technology, Bandung Institute of Technology, Bandung 40132, Indonesia

33421701@mahasiswa.itb.ac.id

Abstract. In the dynamic realm of job shop scheduling (JSS), where decisions regarding the order of job processing have a significant impact on the initial state and performance of the system, addressing the effects of priority changes becomes crucial. To address this challenge, the first part of the study proposes a framework based on digital twin (DT) technology which has the potential to capture the priority changes in JSS, enabling improved decision-making with better insights. In the second part of the study, a robust testing approach has been presented to evaluate the effectiveness of attribute-based priority rules for JSS. The priority rules considered include first in first out (FIFO), last in first out (LIFO), shortest processing time (SPT), longest processing time (LPT), earliest due date (EDD), and shortest remaining processing time (SPRT). Hypothetical data is used to evaluate the performance of these rules in terms of minimizing flow time and wait time, which are crucial for optimizing JSS performance. The study emphasizes the substantial influence of priority rules on JSS performance and also highlights the potential benefits of incorporating DT technology for optimizing scheduling practices.

Keywords: Job Shop Scheduling (JSS), Single Machine Case, Digital Twin (DT), Priority Rules.

1 Introduction

1.1 Background

Scheduling is a decision-making process [1] which is widely studied [2,3] and holds a paramount importance in manufacturing process planning [4], as it allows improving system performance, laying the groundwork for many other shop activities [5].

Job shop scheduling (JSS), one of categories of scheduling, is regarded as one of the most important problems in manufacturing due to its effects on the whole company [6]. JSS involves determining the optimal sequence of operations for each job passing through a set of specified but different machines [7]. JSS has been a major challenge

M. Hartono et al. (eds.), *Proceedings of the 4th International Conference on Informatics, Technology and Engineering 2023 (InCITE 2023)*, Atlantis Highlights in Engineering 21,
https://doi.org/10.2991/978-94-6463-288-0_19

for researchers for over 50 years [3,5] and is still regarded as one of the strong NP-hard problems [9]–[17] and thus poses a significant challenge to researchers [1,3].

JSS is categorized by its dynamic nature, as it is subjected to disruptions such as the machine failure, new jobs arrival and rush orders etc. [13]. These disruptions cause original plan to deviate, reducing the efficiency and quality of scheduled execution [13], [18]. As a result, the previously feasible schedule becomes inefficient; necessitating appropriate changes [18]. To solve these challenges, different approaches has been mentioned in [16], including metaheuristics, dispatching rules, multiagent system, etc. with dispatching rules gaining the importance, especially in dynamic realm.

Priority rules [4] is a technique that calculates priority indices for jobs waiting to be processed by a machine [10,11]. It has a crucial role in the scheduling as priority rules can greatly impact performance of the production process and implementing them can yield various benefits:

- Priority rules are crucial in minimizing costs associated with job waiting times and idle time [21] and they are commonly utilized to address computational costs in scheduling due to their ease of implementation and low complexity [22].
- The use of resources such as machines, equipment, etc. can be maximized, resulting in higher performance and money savings.
- By prioritizing jobs based on due dates, scheduling can ensure that orders are completed and delivered on time, improving customer satisfaction.
- Despite advancements in computer science, simple priority rules still hold relevance in academic and practical scheduling environments [22].

According to [23], priority rules can be categorized into three groups: attribute-based rules (FIFO, LIFO, SPT, SRPT, EDD, slack), dependency-based rules (WinQ), and hybrid rules that combine multiple dispatching rules for priority calculations and job assignments. This study aims to provide a robust testing of attribute-based priority rules to offer valuable insights into behavior of these priority rules under controlled conditions.

1.2 Priority rules and digital twin

Digitalization in manufacturing provides the opportunity for companies to achieve enhanced productivity and efficiency [24]. Recently, digital twin (DT), a core and vital tool that allows the close integration of manufacturing information and physical resources [25], has attracted the attention of many scholars and has been applied into the manufacturing [26]. Using DT in manufacturing provides several benefits, including the integration and visualization of data from manufacturing resources, processes and services, a seamless fusion of physical and digital worlds, etc. [27], which could be used for decision making. The digital twin is acknowledged in literature as a revolutionary technology, that acts as a replica of a real system (for example, product, machine, process, people, etc.) by building a digital replica of physical item [28].

Among the numerous definitions mentioned in [29], one of them is as follows, "*A digital twin is usually a living model with a physical system or asset that continuously adapts operational changes based on collected information and data and can predict* *many physical counterparts' future"*. It can reflect the behavior and real-time state of its physical object accurately so that the manufacturing processes and production operations can be analyzed, monitored, predicted and optimized [25,28].

In the above scenario, DT holds immense potential for optimizing production processes and enabling informed decision-making while priority rules play a crucial role in leveraging the capabilities of DT to optimize production processes and make informed decisions. The integration of priority rules within the DT framework brings together the power of DT and the flexibility of priority rules, resulting in a highly adaptable and agile JSS system.

Through real-time monitoring and tracking of production processes, DT collects valuable data from various sources, such as jobs and machines etc., providing detailed information for effective decision-making where the virtual replica simulates various scheduling scenarios, taking into account the impact of different priority rules. This empowers decision-makers to identify the most optimal schedule and make informed real-time decisions. The implementation of the DT also allows for continuous monitoring and analysis of the shop floor, enabling the timely detection of deviations from the optimal schedule.

Through the use of priority rule-based adjustments, which are computationally efficient, the schedule can be dynamically modified based on the current state of the shop floor. This correction approach significantly improves the effectiveness of the shop floor operations by minimizing deviations and bringing the actual production closer to the optimal schedule. Through this, production systems can be handled [30].

In the above context, [18] provided a framework to capture disturbing events and mitigate their negative impact through DT. It explicitly addresses the priority change of jobs that affect the original schedule and causes deviations. Based on work done in [18], a framework is provided to capture the priority change of jobs (see Fig. 1.)

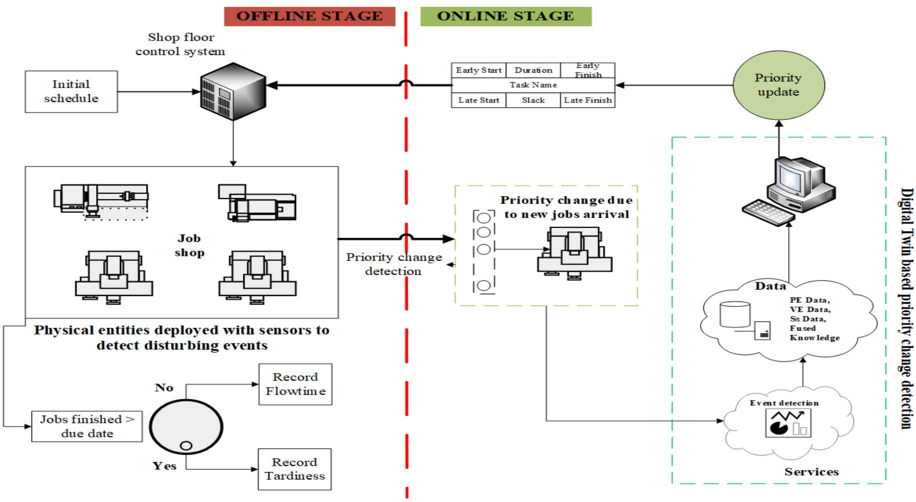

Fig. 1. Framework to detect priority change of jobs through digital twin

1.3 Research gap and approach

According to [23], priority rules can be categorized into three groups: attribute-based rules (FIFO, LIFO, SPT, SRPT, EDD, slack), dependency-based rules (WinQ), and hybrid rules. Significant research has been done on the utilization of these priority rules in JSS (see Table 1), with pieces of research that did comparative studies for different priority rules (for instance, [31]–[33]). To further extend this domain, this study focuses on analyzing the effectiveness of attribute-based priority rules (FIFO, LIFO, SPT, LPT, EDD, SRPT), which stand out for their simplicity and ease of implementation. What makes them unique is their primary use of specific task attributes (e.g., processing time, due date, or remaining processing time etc.) for determining the sequence of jobs execution, impacting production flow, resource allocation, and meeting deadlines. For example, FIFO and LIFO prioritize jobs based on order of jobs arrival time, SPT and LIFO aim at jobs with shorter and longer processing time accordingly, EDD stresses timely completion based on due dates, and SRPT directs attention to jobs with the shortest processing time remaining.

Table 1. Literature study

Priority rules	Literature	This article
FIFO	[4], [7], [23]	✓
LIFO	[4], [7], [34]	✓
EDD	[4]	✓
SPT	[4]	✓
LPT	[4]	✓
SRPT	[31]–[33]	✓

In JSS where newly arriving jobs come up with diverse scenario (e.g., varying processing time and jobs that must be completed efficiently to meet the deadlines, etc.), the importance of these rules could not be disregarded, especially SRPT rule, which is known for its adaptability in such scenarios. By implementing SRPT, shop could respond in a more effective way to the changes in processing times, dynamic jobs arrival, etc. Even though there have been studies which included SRPT as one of priority rule for comparison, the specific effectiveness of SRPT has not directly been addressed.

In this context, we aimed to separate the effects of distinct priority rules on JSS performance by employing robust testing on hypothetical data which offers the opportunity to systematically compare priority rules, developing a better understanding of their effect on waiting time and flow time in a controlled environment. Finally, results were compared that showcased that SRPT performed better than other attribute-based rules.

To initiate our study, we will develop a model for a single machine, driven by three primary reasons; 1- a single machine model serves as a fundamental building block for all industrial scheduling systems, including JSS, 2- the beginning with a single machine model is comparatively simpler, and 3- the solution methods for a single machine could be a basis for developing solution methods for a more complex system. The aim is to present a foundational investigation of performance of attribute-based priority rules under controlled conditions, providing valuable insights into their behavior.

The article proceeds with the formulation of the mathematical model for the single machine environment in Section 2, followed by the methodology involving random data generation in Section 3. Results and discussions are presented in Section 4, while the conclusion is provided in Section 5. The references are listed in Section 6.

2 Mathematical Model

2.1 Problem Description

A set of m jobs $J = \{J_1, J_2, \dots, J_m\}$, have been scheduled on a single machine M and each job J_j consists of a single operation O_{j1}. The aim is to find best sequence of jobs that can be processed on the machine such that the flow time and wait time of jobs are minimized.

2.2 Assumptions

Attributes of jobs, such as arrival time of jobs etc. are not known. Each job is assigned to the machine for processing, and that machine can only process one job at a time. Once a job starts being processed on a machine, it cannot be interrupted. Additionally, there are no setup times required between jobs.

2.3 Model notations

Index
j = job index ($j = 1,2, \dots, m$)

Indices
m = number of jobs

Sets
J = set of jobs
T = set of time periods ($T = \{0,1, \dots, T_{max}\}$)

Parameters
P_j = processing time of job j
C_t = capacity of machine at time t
t = current time

Decision variables
S_{j_t} = starting time of job j at time t
C_{j_t} = completion time of job j at time t

L_{j_t} = binary decision variable at time t
R_{j_t} = remaining processing time of job j

Variables
F_{j_t} = flow time of j at time t
w_{j_t} = waiting time of jobs at time t

2.4 Model formulation

Objective function.
The objective function is to minimize the total flow time of jobs;

$$\text{Minimize } \Sigma F_{j_t}, \forall \text{ j } \in \text{J } and\ t \in T \quad (1)$$

Constraints.

Capacity constraint.
The total processing time of jobs cannot exceed maximum machine capacity;

$$P_j * \text{L}_\text{j} \leq C_t, \forall \text{ j } \in \text{J and t} \in \text{T} \quad (2)$$

Precedence constraint.
The start time of each job must be greater than or equal to the completion time of the previous job plus the waiting time;

$$S_{j_t} \geq \text{C}_{(\text{j}-1)_\text{t}} + P_{\text{j}-1} + w_{j_t} - (C_t \times L_{j_t}), \forall \text{ j } \in \text{J and t} \in \text{T} \quad (3)$$

Non-negative start time.
The start time of each job must be non-negative;

$$S_{j_t} \geq 0, \forall \text{ j } \in \text{J and t} \in \text{T} \quad (4)$$

Flow time calculation.
Since flow time refers to the total time that a job spends in the system, from its release into the system until its completion;

$$F_{j_t} = C_{j_t} - S_{j_t} + P_j, \forall \text{ j } \in \text{J and t} \in \text{T} \quad (5)$$

Completion time.
The completion time of each job plus its processing time must be less than or equal to the sum of the start time of each job plus total processing time. The (6) and (7) ensure that the completion time of job j at time t does not exceed the total processing time of all completed jobs up to time t.

$$C_{j_t} = S_{j_t} + P_j, \forall \text{ j } \in \text{J and t} \in \text{T} \quad (6)$$

$$C_{jt} \leq \Sigma S_{j'_{t'}} + \Sigma P_{j'}, \forall \text{ j, j}' \in \text{J and t, t}' \in \text{T } where\ t' \leq t \tag{7}$$

Wait time calculation.
The waiting time of job j is the difference between the arrival time of the job A_{jt} and the start time of job j is S_{j_t}, with a minimum value of 0.

$$w_{j_t} = \max\left(0, A_{jt} - S_{j_t}\right), \forall \text{ j } \in \text{J where j} > 1 \text{ and t} \in [0, \text{T}] \tag{8}$$

Assignment of jobs.
Each job can only be processed by one machine;

$$\Sigma L_{jt} = 1,\ \forall \text{ j } \in \text{J and t} \in \text{T} \tag{9}$$

Starting time of jobs.
A new job cannot be started until the previous job is finished processing;

$$S_{j_t} \geq S_{(j-1)_t} + P_{j-1}, \forall \text{ j} \in J\ where\ j > 1 \text{ and t} \in \text{T} \tag{10}$$

First in first out (FIFO).
Jobs are processed in the order they arrive at machine. For each job j, the starting time S_{j_t} must be greater than or equal to the completion time of the previous job $C_{(j-1)_t}$:

$$S_{j_t} \geq C_{(j-1)_t}, \forall \text{ j} \in \text{J},\ j \neq 1 \text{ and t} \in \text{T} \tag{11}$$

Last in first out (LIFO).
In LIFO, the last job to arrive at the machine is processed first. For each job j, the starting time S_{j_t} must be greater than or equal to the completion time C_{j_t} of the next job:

$$S_{j_t} \geq C_{(j+1)_t}, \forall \text{ j} \in \text{J}, j \neq m \text{ and t} \in \text{T} \tag{12}$$

Shortest processing time (SPT).
This shows that the jobs with the shortest processing time are processed first. For any two jobs i and j, if i has a shorter P than j, then i must be processed before j:

$$S_{i_t} + P_i \leq S_{j_t}, if\ P_i < P_j, \forall\ i, j \in J, \text{and t} \in \text{T} \tag{13}$$

Longest processing time (LPT).
LPT ensures that jobs with the longest processing time are processed first. For any two jobs i and j, if i has a longer P than j, then i must be processed before j:

$$S_{i_t} + P_i \geq S_{j_t} + P_j, if\ P_i > P_j, \forall\ i, j \in J \text{ and t} \in \text{T} \tag{14}$$

Shortest remaining processing time (SRPT).
This constraint ensures that jobs with the shortest remaining processing time are processed first. We add a new decision variable R_{j_t}, which represents the remaining processing time of job j. It is defined as:

$$R_{j_t} = P_j - \left(t - S_{j_t}\right), if\ S_{j_t} \le t, \forall j \in J \text{ and } t \in T \quad (15)$$

$$R_{j_t} = P_j\ , if\ S_{j_t} > t, \forall\ j \in J \text{ and } t \in T \quad (16)$$

For any two jobs i and j, if i has a shorter remaining processing time than j, then i must be processed before j:

$$S_{i_t} + R_{i_t} \le S_{j_t} + R_{j_t}\ , if\ R_{i_t} < R_{j_t}, \forall\ i, j \in J \text{ and } t \in T \quad (17)$$

3 Methodology

3.1 Random data generation

In the study, MS Excel was used to generate data that involve variability and uncertainty in job arrivals and processing times (see Fig. 2 and Table 2). To generate the data, parameters such as arrival times, processing times, and due dates were used. Based on these parameters, data for 20 jobs was randomly generated.

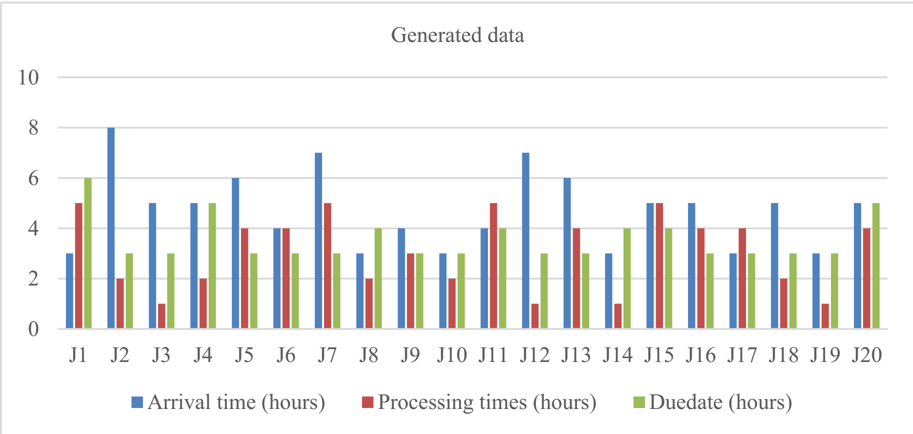

Fig. 2. Graph showing values for arrival time, processing time and due dates for 20 jobs

Table 2. Hypothetical data generated

Jobs ID	Arrival time (hours)	Processing time (hours)	Due date (hours)
J_1	3	5	6
J_2	8	2	3
J_3	5	1	3
J_4	5	2	5
J_5	6	4	3
J_6	4	4	3
J_7	7	5	3
J_8	3	2	4
J_9	4	3	3
J_{10}	3	2	3
J_{11}	4	5	4
J_{12}	7	1	3
J_{13}	6	4	3
J_{14}	3	1	4
J_{15}	5	5	4
J_{16}	5	4	3
J_{17}	3	4	3
J_{18}	5	2	3
J_{19}	3	1	3
J_{20}	5	4	5

3.2 Calculations based performance metrices

To validate the model, LINGO and Microsoft Excel were utilized. LINGO ensured model accuracy and completeness, while Excel facilitated the application of different priority rules. By using the following method, we evaluated and compared different priority rules and determined their impact on JSS against flow and wait time:

- Jobs sequence: to determine the order in which jobs will be processed, we applied the chosen priority rule (e.g., FIFO, LIFO, SPT, LPT, EDD, SRPT).
- Start time: It was determined based on the completion time of the preceding job. The first job would start at its arrival time ($S_j = A_j$, when j $\in 1, S_j = C_{j-1}, \forall$ j > 1).
- Finish time: Finish time of each job was calculated by adding its processing time to its start time ($F_j = S_j + P_j$, $\forall$ j $\in$ J).
- Wait time: The wait time of each job was determined by calculating the difference between its start time and arrival time ($w_j = S_j - A_j, \forall$ j $\in$ J).
- Flow time: We calculated the flow time of each job as the difference between its completion time and arrival time ($FT_j = F_j - A_j, \forall$ j $\in$ J).
- Total flow time and total wait time: The flow times and wait times of all jobs were summed up to obtain the total flow time ($TFT_{priority\_rule} = \Sigma F_j$, $\forall$ j $\in$ J) and total wait time ($Tw_{priority\_rule} = \mathrm{w_j}$, $\forall$ j $\in$ J).

Scenario I: first in first out (FIFO)

- According to the FIFO, the sequence of the jobs is based on the arrival times, where the first job entering the system will be processed first and latest will be processed latest, while minimizing the flow and wait time of jobs, reducing their time in the system. For FIFO, the job sequence is as follows; $J_8 - J_1 - J_{14} - J_{10} - J_{19} - J_{17} - J_9 - J_6 - J_{11} - J_{16} - J_3 - J_4 - J_{15} - J_{18} - J_{20} - J_5 - J_{13} - J_{12} - J_7 - J_2$.
- The start time, finish time, wait time, flow time, and total wait time for each job were calculated using the formulas described in section 3.2.
- Additionally, the total flow time and the total wait time of all jobs were determined; Total flow time=570 and total wait time=505. See Table 3 for calculations.

Table 3. Calculations for FIFO rule

Jobs ID	Arrival time (hours)	Pro-cessing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_8	3	1	4	3	4	0	1
J_1	3	1	3	4	5	1	2
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_7	7	5	3	61	66	54	59
J_2	8	2	3	66	68	58	60

Scenario II: Last in first out (LIFO)

- According to LIFO, last job entering to the system will be processed first, minimizing the wait time of jobs. For LIFO, the sequence is as follows; $J_2 - J_7 - J_{12} - J_{13} - J_5 - J_{20} - J_{18} - J_{15} - J_4 - J_3 - J_{16} - J_{11} - J_9 - J_6 - J_{17} - J_{19} - J_{10} - J_{14} - J_1 - J_8$.
- The time calculations (start, finish, wait, flow and wait) for each job, were calculated according to formulas mentioned in section 3.2.
- Total flow time = 728 and total wait time = 667. See Table 4 for calculations.

Table 4. Calculations for LIFO rule

Jobs ID	Arrival time (hours)	Pro-cessing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_2	8	2	3	8	10	0	2
J_7	7	5	3	10	15	3	8
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_1	3	5	6	62	67	59	64
J_8	3	2	4	67	69	64	66

Scenario III: Shortest processing time (SPT)

- Jobs sequence: As per SPT rule, the job with shortest total processing time [33] will be processed first [32], minimizing the mean flow and wait time. The sequence is as follows, $J_{12} - J_3 - J_{19} - J_{14} - J_{18} - J_2 - J_8 - J_{10} - J_4 - J_9 - J_6 - J_5 - J_{20} - J_{17} - J_{13} - J_{16} - J_{11} - J_1 - J_7 - J_{15}$.
- The time calculations (start, finish, wait, flow and wait) for each job, were calculated according to formulas mentioned in section 3.2.
- Total flow time = 523 and total wait time = 462. See Table 5 for calculations.

Table 5. Calculations for SPT rule

Jobs ID	Arrival time (hours)	Pro-cessing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_{12}	7	1	3	7	8	0	1
J_3	5	1	3	8	9	3	4
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_7	7	5	3	58	63	51	56
J_{15}	5	5	4	63	68	58	63

Scenario IV: Longest processing time (LPT)

- Job sequence: As per LPT rule, the job with longest total processing time [33] will be processed first [32]. The sequence is as follows, $J_{15} - J_7 - J_1 - J_{11} - J_{16} - J_{13} - J_{17} - J_{17} - J_{20} - J_5 - J_6 - J_9 - J_4 - J_{10} - J_8 - J_2 - J_{18} - J_{14} - J_{19} - J_3 - J_{12}$.
- The time calculations (start, finish, wait, flow and wait) for each job, were calculated according to formulas mentioned in section 3.2.
- The total flow time = 810 and total wait time = 749. See Table 6 for calculations.

Table 6. Calculations for LPT rule

Jobs ID	Arrival time (hours)	Pro-cessing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_{15}	5	5	3	5	10	0	5
J_7	7	5	3	10	15	3	8
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_3	5	1	3	64	65	59	60
J_{12}	7	1	4	65	66	58	59

Scenario 5: Earliest due date (EDD)

- Jobs sequence: As per EDD rule: the job with earliest due date will be processed first, minimizing mean flow and wait time by ensuring on-time completion. The sequence is as follows, $J_3 - J_7 - J_2 - J_5 - J_7 - J_{10} - J_{12} - J_{13} - J_9 - J_{19} - J_{16} - J_{17} - J_{18} - J_{15} - J_8 - J_{11} - J_{14} - J_4 - J_{20} - J_1$.
- The time calculations (start, finish, wait, flow and wait) for each job, were calculated according to formulas mentioned in section 3.2.
- The total flow time = 651 and total wait time = 589. See Table 7 for calculations.

Table 7. Calculations for EDD rule

Jobs ID	Arrival time (hours)	Processing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_3	5	1	3	5	6	0	1
J_7	7	5	3	7	12	0	5
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_{20}	5	4	5	59	63	54	58
J_1	3	5	6	63	68	60	65

Scenario 6: Shortest remaining processing time (SRPT)

- Jobs sequence: As per SRPT rule, the job with least total remaining processing time will be processed first [33,34], effectively minimizing mean flow and wait time. The sequence is as follows; $J_3 - J_{19} - J_{14} - J_8 - J_{10} - J_2 - J_4 - J_{18} - J_9 - J_{12} - J_{13} - J_{16} - J_7 - J_{11} - J_{15} - J_6 - J_5 - J_{17} - J_{20} - J_1$.
- The time calculations (start, finish, wait, flow and wait) for each job, were calculated according to formulas mentioned in section 3.2.
- The total flow time = 395 and total wait time = 334. See Table 8 for calculations.

Table 8. Calculations for SRPT rule

Jobs ID	Arrival time (hours)	Processing time (hours)	Due date (hours)	Start time (hours)	Finish time (hours)	Waiting time (hours)	flow Time (hours)
J_3	5	1	3	5	6	15	16
J_{19}	3	1	3	6	7	2	3
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
J_{20}	5	4	5	57	61	25	29
J_1	3	5	6	61	66	57	62

4 Results and Discussions

To determine the best priority rule, the total flow time and the total wait time were used as performance measures. Based on the results obtained from the study, the total flow time and total wait time for each priority rule are shown in Table 9 and Fig. 3:

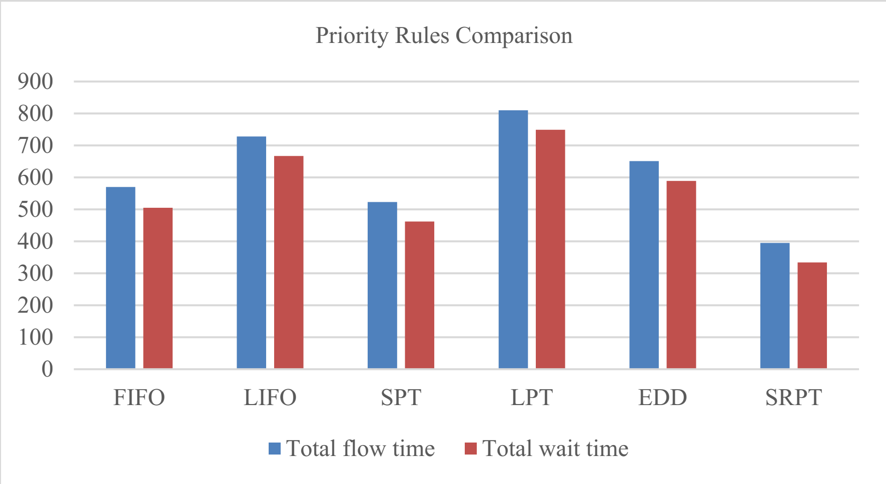

Fig. 3. Comparison of priority rules

Table 9. Results comparison for priority rules

Priority rule	Total flow time	Total wait time
FIFO	570	505
LIFO	728	667
SPT	523	462
LPT	810	749
EDD	651	589
SRPT	395	334

The total flow time represents the overall time it takes for all jobs to complete. A lower total flow time indicates a more efficient scheduling approach where jobs are completed in less time. In this study, the SRPT rule had the lowest total flow time of 395, followed by the SPT rule with a total flow time of 523. Both of these rules outperformed the other four rules in terms of minimizing the overall flowtime of the jobs.

The total wait time measures the waiting time experienced by each job before it begins processing. A lower total wait time indicates that jobs spend less time waiting and can start processing sooner. In this study, the SRPT rule again performed the best with the lowest total wait time of 334. The SPT rule also had a relatively low total wait time of 462. SRPT showed less flow time and less total wait time among all the tested rules.

5 Conclusion

In conclusion, the study highlights the potential benefits of incorporating DT technology for optimizing scheduling practices and achieving efficient production outcomes. For this, a framework based on DT, to capture priority change of jobs, has been provided first. Then the study evaluated the performance of different priority rules in a JSS system. The priority rules evaluated were FIFO, LIFO, SPT, LPT, EDD, and SRPT. The results indicated that the SRPT rule achieved the best performance in terms of both total flow time and total wait time. This suggests that the SRPT rule is effective in minimizing job flow time and reducing waiting time for jobs in the system.

Looking ahead, the model can be extended to a JSS system. The integration of more priority rules with DT could be explored to further enhance the scheduling problems. Also, a virtual JSS with the DT in place while modeling various scheduling scenarios could be employed while accounting for the impact of different priority rules. Moreover, by utilizing DT, real-time data and insights can be incorporated, allowing for dynamic adjustments and improved decision-making. This integration has the potential to optimize scheduling strategies and improve the overall performance of the production system.

References

1. Barták, R. Vlk, M.: Reactive recovery from machine breakdown in production scheduling with temporal distance and resource constraints. In : ICAART 2015 - 7th International Conference on Agents and Artificial Intelligence Proceedings, vol. 2. SCITEPRESS - Science and Technology Publications, Lisbon (2015).
2. Balasundaram, A. K.: A Simulation Modeling Approach for Job Shop Scheduling Problems: Case of Metal Industry. International Journal of Science and Research (IJSR) 7(10), 421-425 (2018).
3. Bożejko, W., Rajba, P., Wodecki, M.: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics. Springer, Heidelberg (2020).
4. Palit, T., Islam, M.: Development of an Arena Simulation Model for Scheduling Problems in Job Shop Production. In : ICMIME 2019. RUET, Rajshahi (2019).
5. Wu, L. H., Chen, X., Chen, X. D., Chen, Q. X.: The research on proactive-reactive scheduling framework based on real-time manufacturing information. Materials Science Forum 626-627, 789-794 (2009).
6. Shojaeinasab, A., Charter, T., Jalayer, M., Khadivi, M., Ogunfowora, O., Raiyani, N., Yaghoubi, M., Najjaran, H.: Intelligent manufacturing execution systems: A systematic review. Journal of Manufacturing Systems 62 503-522 (2022).
7. Shahbazi, S., Sajadi, S. M., Jolai, F.: A Simulation-Based Optimization Model for Scheduling New Product Development Projects in Research and Development Centers. Iranian Journal of Management Studies 10(4), 883-904 (2017).
8. Nawara, P. G. M.: A Simulation Model for the Real-Life Job-Shop Scheduling Problem. International Journal of Engineering Innovation & Research 2(2), 188-191 (2013).

9. Hassam, A., Keddari, N., Ain Sebaa, F., Sari, Z.:Comparative study between methods for selection alternative routings in real time. In : 2011 International Conference on Communications, Computing and Control Applications, pp. 1-4. IEEE, Tunisia (2011).
10. Hassanein, W., Nawara, G. M., Wael Hassanein, E. S.: Solving the Job-Shop Scheduling Problem by Arena Simulation Software Productivity View project Solving the Job-Shop Scheduling Problem by Arena Simulation Software. International Journal of Engineering Innovation & Research 2(2), 161-166 (2013).
11. Zupan, H., Žerovnik, J., Herakovič, N.: Service Orientation in Holonic and Multi-Agent Manufacturing. Springer, Heidelberg (2018).
12. Laosiritaworn, W., Pinchai, N., Jarupuncharadsri, P.: Job Shop Layout Improvement with Simulation Technique: A Case of Metal Sheet Production. IOP Conference Series: Materials Science and Engineering 842, 012023 (2020).
13. Wang, M., Zhang, P., Zheng, P., He, J., Zhang, J., Bao, J.: An Improved Genetic Algorithm with Local Search for Dynamic Job Shop Scheduling Problem. In : 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), pp. 766-771. IEEE: Hong Kong (2020).
14. Liaqait, R. A., Hamid, S., Warsi, S. S., Khalid, A.: A critical analysis of job shop scheduling in context of Industry 4.0. Sustainability (Switzerland) 13(14), 7684 (2021).
15. Leusin, M., Frazzon, E., Uriona Maldonado, M., Kück, M., Freitag, M., Solving the Job-Shop Scheduling Problem in the Industry 4.0 Era. Technologies (Basel) 6(4), 107 (2018).
16. Zhang, L., Gao, L. Li, X.: A hybrid intelligent algorithm and rescheduling technique for job shop scheduling problems with disruptions. International Journal of Advanced Manufacturing Technology 65, 1141-1156 (2013).
17. Chaudhry, I. A., Khan, A. A.: A research survey: Review of flexible job shop scheduling techniques. International Transactions in Operational Research 23(3), 551-591 (2016).
18. Usman Nisar, M., Ma'ruf, A., Halim, A. H.: Development Of Framework For Flexible Job Shop Scheduling Based On Digital Twin To Tackle Disturbing Events.
19. Dominic, P. D. D., Kaliyamoorthy, S., Kumar, M. S.: Efficient dispatching rules for dynamic job shop scheduling. International Journal of Advanced Manufacturing Technology 24, 70-75 (2004).
20. Singh, R., Mathirajan, M.: Experimental investigation for performance assessment of scheduling policies in semiconductor wafer fabrication—a simulation approach. International Journal of Advanced Manufacturing Technology 99 (5–8), 1503–1520 (2018).
21. Kumar, K. K., Nagaraju, D., Gayathri, S., Narayanan, S.: Evaluation and Selection of Best Priority Sequencing Rule in Job Shop Scheduling using Hybrid MCDM Technique. IOP Conference Series: Materials Science and Engineering 197, 012059 (2017).
22. Sels, V., Gheysen, N., Vanhoucke, M.: A comparison of priority rules for the job shop scheduling problem under different flow time- and tardiness-related objective functions. International Journal of Production Research 50(15), 4255-4270 (2012).
23. Turker, A. K., Aktepe, A., Inal, A. F., Ersoz, O. O., Das, G. S., Birgoren, B., A decision support system for dynamic job-shop scheduling using real-time data with simulation. Mathematics 7(3), 278 (2019).
24. Kamble, S. S., Gunasekaran, A., Parekh, H., Mani, V., Belhadi, A., Sharma, R., Digital twin for sustainable manufacturing supply chains: Current trends, future perspectives, and an implementation framework. Technological Forecasting and Social Change 176, 121448 (2022).
25. Bao, J., Guo, D., Li, J., Zhang, J.: The modelling and operations for the digital twin in the context of manufacturing. Enterprise Information Systems 13(4), 534-556 (2019).

26. Y. Li, Z. Tao, L. Wang, B. Du, J. Guo, and S. Pang, “Digital twin-based job shop anomaly detection and dynamic scheduling,” *Robot Comput Integr Manuf*, vol. 79, 2023, doi: 10.1016/j.rcim.2022.102443.
27. C. Zhang, W. Xu, J. Liu, Z. Liu, Z. Zhou, and D. T. Pham, “Digital twin-enabled reconfigurable modeling for smart manufacturing systems,” *Int J Comput Integr Manuf*, vol. 34, no. 7–8, 2021, doi: 10.1080/0951192X.2019.1699256.
28. A. Corallo, V. Del Vecchio, M. Lezzi, and P. Morciano, “Shop floor digital twin in smart manufacturing: A systematic literature review,” *Sustainability (Switzerland)*, vol. 13, no. 23. MDPI, Dec. 01, 2021. doi: 10.3390/su132312987.
29. S. Mathupriya, S. Saira Banu, S. Sridhar, and B. Arthi, “WITHDRAWN: Digital twin technology on IoT, industries & other smart environments: A survey,” *Mater Today Proc*, 2020, doi: 10.1016/j.matpr.2020.11.358.
30. M. Saqlain, S. Ali, and J. Y. Lee, “A Monte-Carlo tree search algorithm for the flexible job-shop scheduling in manufacturing systems,” *Flex Serv Manuf J*, 2022, doi: 10.1007/s10696-021-09437-4.
31. L. E. Schrage and L. W. Miller, “ The Queue M / G /1 with the Shortest Remaining Processing Time Discipline ,” *Oper Res*, vol. 14, no. 4, 1966, doi: 10.1287/opre.14.4.670.
32. N. Kundakci and O. Kulak, “Hybrid genetic algorithms for minimizing makespan in dynamic job shop scheduling problem,” *Comput Ind Eng*, vol. 96, 2016, doi: 10.1016/j.cie.2016.03.011.
33. S. G. Ponnambalam, P. Aravindan, and P. Sreenivasa Rao, “Comparative evaluation of genetic algorithms for job-shop scheduling,” *Production Planning and Control*, vol. 12, no. 6, 2001, doi: 10.1080/095372801750397680.
34. A. Mukid, M. Mukaddes, K. Islam, and T. Palit, “DEVELOPMENT OF AN ARENA SIMULATION MODEL FOR SCHEDULING PROBLEMS IN JOB SHOP PRODUCTION Inventory Optimization using (R, S) Policy for retail chain store View project DEVELOPMENT OF AN ARENA SIMULATION MODEL FOR SCHEDULING PROBLEMS IN JOB SHOP PRODUCTION.” [Online]. Available: https://www.researchgate.net/publication/358493784