Journal of Project Management 9 (2024) 101–108

Contents lists available at GrowingScience

Journal of Project Management

homepage: www.GrowingScience.com

No idle flow shop scheduling models with separated set-up times and concept of job weightage to optimize rental cost of machines

Shakuntala Singla[a], Harshleen Kaur[b*], Deepak Gupta[c] and Jatinder Kaur[d]

[a]*Associate Professor (Mathematics), M.M.E.C., Maharishi Markandeshwar Deemed to Be University, Mullana, Ambala, Haryana, India*
[b]*Research Scholar (Mathematics), M.M.E.C., Maharishi Markandeshwar Deemed to Be University, Mullana, Ambala, Haryana, India*
[c]*Professor & Head of Department Mathematics, M.M.E.C., Maharishi Markandeshwar Deemed to Be University, Mullana, Ambala, Haryana, India*
[d]*Associate Professor & Head of Department Mathematics, Guru Nanak Girls College, Yamuna Nagar, Haryana, India*

C H R O N I C L E

Article history:
Received: November 20, 2023
Received in revised format: December 22, 2023
Accepted: February 4, 2024
Available online:
February 4, 2024

Keywords:
Flowshop
Set-up time
No idle
Sequence
Scheduling
Weightage

A B S T R A C T

The current paper investigates a two-stage flow shop scheduling model with no idle restriction, in which the time taken by machines to set-up is separately considered from the processing time. Owing to inherent usefulness as well as relevance in real-world situations, jobs' weight has additionally included. To eliminate machine idle time and cutting machine cost of rental, the reason for the conduct of the study is to provide a heuristic algorithm which, once put into practice, processes jobs in an optimal way, guarantees in smallest conceivable make span. Multiple computational examples generated in MATLAB 2019a serve as testament to the efficacy of the proposed strategy. The outcomes are contrasted with the current methods that Johnson, Palmer and NEH have demonstrated.

1. Introduction

The process of scheduling is an essential and integral aspect of resource allocation, wherein the deployment of assets is carefully planned and executed to facilitate the execution of activities. The chief goal of scheduling is in order to identify the most optimal solution, taking into contemplation the pressing desire for optimum a specific purpose or outcome. The well-known flow shop scheduling problem(FSSP) involves evaluating the best sequence for two or more jobs to be performed on two or more pre-ordered machines to optimize some measure of effectiveness. The critical constraint in an industrialized flow shop scenario is the no-idle time on machines or the inability to halt a machine after it has been started. As a result, there can be no downtime for the machines as they must run continually. Significant emphasis was devoted to resolving the scheduling problem over the past half-century. In the realm of flow shop scheduling problems, Johnson(1954) is credited with pioneering the development of a groundbreaking mathematical model. This model, which marked a significant milestone in the field, achieved an optimum solution as a remarkable success. The effectiveness of Johnson's notion grabs considerable interest among multiple scholars, who have a propensity to investigate this tactic. To reduce the make-span, Palmer (1985) applied the heuristic technique for the problems characterized by a set of n-job m-machine. The NEH method has since gained significant attention in the field of scheduling problems. The NEH method focuses on optimizing the scheduling of tasks across multiple machines to achieve a reduction in the overall processing time (Nawaz et al., 1983). In the realm of research, (Jackson, 1956; Ignall, 1965; Campbell et al., 1970) as well as (Gupta & Shashi, 2012) have made significant contributions by developing upon their initial investigations.

* Corresponding author.
E-mail address: harshleenkaur34953@gmail.com (H. Kaur)

doi: 10.5267/j.jpm.2024.2.001

The absence of arguments on the idea of job weightage in scheduling models before 1980 is a notable observation derived from Johnson's pioneering research in 1954. In a two-stage flow shop scheduling method where the processing time is linked to probabilities, including job blocks, they optimized the weighted mean rental cost (Miyazaki & Nishiyama, 1980). A remedy for the 2-machine s, n-jobs flow shop scheduling problem with the intent to further improve the weighted mean flow time of jobs was established (Maggu & Das, 1982).

It is widely acknowledged that setup times pose significant challenges and are often considered as one of the most prevalent complicating factors in scheduling tasks. The pioneering work of Yoshida and Hitomi (1979) marked the initiation of research into the flow shop scheduling problem, specifically focusing on the separation of setup times from processing times. Building upon Johnson's rule, they introduced an extension that allowed for a more comprehensive analysis. Authors' model (Kim & Bobrowski, 1994) utilizes computer simulation techniques by incorporating sequence-dependent setup times to analyze and optimize the scheduling process in a limited machine job shop. The model captures the realistic dynamics of job sequencing and setup operations. By examining these factors, the study sought to provide valuable insights into the complexities of this problem and potentially offer novel solutions or approaches for its resolution.(Allahverdi et al., 1999).

No-idle flow shop scheduling entails no-idle constraints, which means that machines constantly operate with no breaks. The first investigation of the m-machine no-idle condition in a flow shop was conducted (Adiri & Pohoryles, 1982). In recent years, the burgeoning interest in solving scheduling problems has been notably directed towards meta-heuristics. This traction is owed to their capacity to yield high-caliber solutions while maintaining computational efficiency. Moreover, within the sphere of pertinent literature, Pan and Wang (2008a, 2008b) have introduced discrete differential evolution (DDE) and discrete particle swarm optimization (DPSO) algorithms tailored for analogous problem domains. These papers delineate a novel speed-up scheme specifically addressing the insertion neighborhood. This scheme significantly mitigates the computational complexity associated with a singular insertion neighborhood scan, reducing it from $O(n^3m)$ to $O(n^2m)$ when the insertion sequence is adhered to. This acceleration methodology draws inspiration from the techniques elucidated by Taillard (1990), specifically tailored for analogous neighborhood searches, albeit in the context of the traditional flow shop problem. Both algorithms employed an advanced local search technique known as Iterated Greedy (RUIZ & STÜTZLE, 2007). The evaluation of these, both DDE and DPSO algorithms leveraged the renowned benchmark suite introduced by Taillard (1993), treating the instances therein as instances of the classical flowshop for the no idle problem, thereby facilitating a rigorous assessment of their efficacy. The comparative analysis in the two studies involved an assessment of the proposed methodologies against the heuristics proposed by Baraz and Mosheiov (2008) and Kalczynski and Kamburowski (2005).

Concluding the investigation, Ruiz et al. (2009) introduced an Iterated Greedy (IG) algorithm tailored for the No Idle Permutation Flowshop (NIPFS) problem, emphasizing the makespan criterion. Their investigation involved the establishment of a proprietary benchmark standard, facilitating a comprehensive performance evaluation of IG in contrast to existing heuristics and meta-heuristics from the relevant literature. Among the heuristics scrutinized, the authors draw attention to two specifically adapted for the NIPFS problem: FRB3 proposed by Rad et al. (2009) and GH_BM2, incorporating accelerations derived from the two phases of GH_BM introduced by Baraz and Mosheiov (2008). Goncharov and Sevastyanov(2009) introduced a collection of polynomial time heuristics grounded in a geometric approach specifically tailored for addressing the Fm/no-idle/C_{max} problem, with a focus on instances involving 3 and 4 machines. Additionally, the authors conducted an extensive survey encompassing pertinent literature, providing a comprehensive overview of works relevant to their proposed methodologies. An algorithm employing differential evolution with a variable parameter search (vpsDE) is developed and juxtaposed against a widely recognized random key genetic algorithm (RKGA) documented in the literature. Deng and Gu (2012) introduced a hybrid discrete differential evolution (HDDE) algorithm specifically tailored for addressing the Fm/no-idle/C_{max} problem. They also proposed an innovative speed-up method leveraging network representation to assess the entire insert neighborhood of a job permutation, subsequently integrated into HDDE. Furthermore, an effective modification to the insert neighborhood local search in HDDE was implemented to strike a balance between global exploration and local exploitation. The experimental findings demonstrate the superior performance of HDDE compared to existing state-of-the-art algorithms. Zhou et al.(2014) introduced an invasive weed-optimal scheduling algorithm designed to optimize the No-Idle Flowshop Scheduling Problem (NFSP). The efficacy of the algorithm was substantiated through a comparative analysis involving 12 instances of varying sizes, demonstrating its effectiveness in addressing the no-idle flowshop scheduling problem when contrasted with alternative algorithms. Yazdani and Naderi (2016) examined the scheduling problem entailing a hybrid flowshop configuration without idle time. They employed mixed-integer linear programming to formally model and formulate the intricacies of the problem. Shao et al. (2018) endeavored to enhance completion time reduction by integrating a memetic algorithm with an edge histogram approach. This integration was implemented as a strategic means to optimize the efficiency of the underlying processes associated with the studied system. The utilization of a memetic algorithm, combined with edge histogram techniques, was carefully orchestrated to synergistically address and mitigate challenges related to completion time within the framework of their research.

While previous research has extensively delved into the no-idle flow shop scheduling problem, predominantly emphasizing completion time (makespan) minimization as the primary objective function, there is a noticeable dearth of studies addressing the associated energy consumption concerns. Addressing this gap, Chen et al. (2019) investigated the distributed no-idle permutation flow-shop scheduling problem. They introduced a collaborative optimization algorithm designed to ensure both diversity and quality within the initial population through the integration of two heuristic synergies. Additionally, enhancements were made to the search operator adaptive reinforcement strategy. The study culminated in the substantiation of the algorithm's effectiveness through numerical experiments of varying scales, demonstrating its capability to address the intricacies of the aforementioned scheduling problem.

Particle Swarm Optimization (PSO) stands as a representative swarm intelligence algorithm, valued for its uncomplicated structure and rapid computational efficiency, making it a common choice for tackling various NP-hard problems. However, when applied to the Discrete Flow Shop Scheduling Problem (DFSP), characterized by multiple constraints, the limitations of single-objective PSO become apparent. In response to this challenge, researchers have explored Multiobjective Particle Swarm Optimization (MoPSO), a methodology tailored for addressing problems with multiple objectives, offering greater practicality than traditional PSO. Given that an excessive number of constraints in DFSP can detrimentally impact algorithmic performance, there is a recognized need for further enhancements to MoPSO. In pursuit of this objective, Zhang et al.(2021) introduced a refined version of MoPSO specifically designed to enhance the efficiency of solving a bi-objective mixed no-idle Flow Shop Scheduling Problem (FSP). The problem involves minimizing both the makespan and total processing time. The proposed enhancement involves a novel multiobjective particle swarm optimization technique incorporating multi-directional updates to optimize the solution process.

While taking job weighting into account, Kaur et al. (2021) came up with a way to lower the expense of hiring for the no idle two-stage flow shop scheduling problem. In their study, Singla et al. (2023) encountered an innovative methodology for limiting leasing expenses in the context of no idle two-stage flow shop scheduling. By integrating weightage & transit time factors into the scheduling process, the researchers (Singla, Kaur, Gupta, et al., 2023a) aimed to optimize the allocation of resources and minimize overall rental expenses. The natural world serves as a vast reservoir of knowledge, inspiring organisms to seek solutions to their complex quandaries. Furthermore, scholars and experts have effectively employed this acquired knowledge to address intricate engineering dilemmas (Singla, Kaur, Gupta, et al., 2023b). The statistical optimization maneuvers in question have been extensively explored and documented in various scholarly works. Notably, researchers (Kumari et al., 2021) have made substantial contributions to the existing literature in this domain.

Also, authors of this publication are reaching out to a wider audience by including the set-up times for jobs, building on the Gupta (2021) research. This existing study is centred around the recognition of the finest optimum sequencing of jobs with the objective of lessening expenses associated with the rental of high-cost machinery.

2. Practical situation

The presence of various experimental and practical circumstances is commonly observed throughout everyday involvement in manufacturing and fabrication settings. These scenarios often require the execution of diverse tasks that involve the utilization of different types of industrial equipment. The weightage of jobs can be observed in various industries, including the cotton industry, leather manufacturing unit, and textile factory. These industries serve as practical examples to understand the significance of different job roles and their contributions. Different varieties of cotton, shoes, jackets, and fabric of varying sizes or qualities are carried out in diverse manufacturing facilities, reflecting the diverse range of consumer preferences and market demands. Due to a lack of finances in his early profession, one needs to rent the machines. For example, to start a pathology laboratory, much expensive equipment like a microscope, water bath, lab incubator, glucometer, blood cell counter, tissue diagnostics, etc., one does not buy these machines but instead take on rent. Renting enables saving capital investments, helping choose the right equipment for the job and access the latest technology.

Notations

i	1, 2,...n sequence of jobs
s_1	Sequence optimization employing Johnson's method
h_{i1}	First machine's i-th job processing time
h_{i2}	Second machine's i-th job processing time
S_{i1}	Setup time of first machine H_1
S_{i2}	Setup time of second machine H_2
T_{i2}	Second machine's i-th job completion time
W_i	Weightage of i-th job
$u_1(s_1)$	The time period of machine H_1's utilization within sequence s_1
$u_2(s_1)$	The time period of machine H_2's utilization within sequence s_1
c_1	Time-based fees for rental of machine H_1
c_2	Time-based fees for rental of machine H_2
l_2	To eliminate idle time, the latest time to lease machine H_2
$r(s_1)$	Rental cost for sequence s_1

2.1 Assumptions

- There is no room for any kind of transfer between two different machines, H_1 and H_2, because of processing of jobs which work autonomously in sequential H_1 H_2.
- Simultaneous processing of a single job by two machines is not feasible.
- Any alteration to the machines' path of action is strictly prohibited until the completion of said job becomes unattainable.

- Time spent for setting up and equipment break down are not factored into utilization calculations.

2.2 Rental Policy

The machines are rented on as needed basis and subsequently return them once they are no longer necessary. Specifically, the initial machine acquired through a rental agreement at the commencement of job processing. Subsequently, the second machine will be obtained on a rental basis once the initial job on the first machine has been completed.

3. Problem formulation

Consider the processing of jobs i (where i ranges from 1 to n) by two machines, denoted as H1 and H2. Consider the processing time that are separated from setup times Si1 and Si2 on the machines H1 and H2 denoted by hi1 and hi2 correspondingly. The model's mathematical representation can be expressed mathematically in the form of Table 1 in a matrix-based format. In order to minimize capital expenditures for rented equipment, our mission is to pinpoint the optimum jobs {s1} sequence.

Table 1
Mathematical formulation in a matrix format

Job	Machine H_1		Machine H_2		Weight
i	h_{i1}	S_{i1}	h_{i2}	S_{i2}	W_i
1	h_{11}	S_{11}	h_{12}	S_{12}	W_1
2	h_{21}	S_{21}	h_{22}	S_{22}	W_2
3	h_{31}	S_{31}	h_{32}	S_{32}	W_3
..	..	..	..	..	..
n	h_{n1}	S_{n1}	h_{n2}	S_{n2}	W_n

5. Algorithm

Step 1: Determine the processing times, named as H_{i1}& H_{i2}, for the machines H_1& H_2 respectively:

$$H_{i1} = h_{i1} - S_{i2} \tag{1}$$
$$H_{i2} = h_{i2} - S_{i1} \tag{2}$$

Step 2: For machines H_1& H_2, use the following equation to determine their respective weighted flow times H'_{i1} and H'_{i2} :

(a) If min (H_{i1} , H_{i2}) = H_{i1} , then

$$H'_{i1} = \frac{H_{i1}-W_i}{W_i} \tag{3}$$

and $H'_{i2} = \frac{H_{i2}}{W_i}$

(b) If min (H_{i1} , H_{i2}) = H_{i2} , then

$$H'_{i1} = \frac{H_{i1}}{\mathrm{W_i}} \tag{4}$$

and $H'_{i2} = \frac{H_{i2}+W_i}{W_i}$

Step 3: While cutting down on the total amount of time elapsed, implement on Johnson's method (1954) to acquire the optimum string s_1.
Step 4: For computing the total elapsed time for string s_1, build a flow in-out table.
Step 5: Determine

$$l_2 = T_{i2} - \sum_{n=1}^{\infty} H_{i2} \tag{7}$$

Step 6: In order for machine H_2 to commence processing, the most recent time l_2 considered as the starting point for processing will be employed to generate a flow in-flow out table.

Step 7: Calculate utilization time $u_1(s_1)$ and $u_2(s_1)$ of machines H_1& H_2 by:

$$u_1(s_1) = \sum_{n=1}^{\infty} H_{i1} \tag{8}$$
$$u_2(s_1) = T_{i2} - l_2 \tag{9}$$

Step 8: Finally, calculate

$$r(s_1) = u_1(s_1) * c_1 + u_2(s_1) * c_2 \tag{10}$$

6. Numerical illustration

Taking into consideration, where processing durations separating to the setup times and job weightage are specified in Table 2, assume five jobs and two machines. Four and six units of time are needed to hire machines H1 and H2, respectively. Our goal is to achieve optimal efficiency of sequencing jobs for execution on machines that may be rented for the most economical cost.

Table 2
Problem-specific data set

Jobs	Machine H_1		Machine H_2		Weight
i	h_{i1}	S_{i1}	h_{i2}	S_{i2}	W_i
1	14	2	29	5	3
2	29	6	31	9	5
3	30	5	27	4	2
4	9	1	5	7	1
5	12	3	8	2	4

Solution: In accordance with Step 1, Table 3 presents an overview of the anticipated processing times on machines H_1 and H_2. Following Step 2, Table 4 displays weighted flow shop times H'_{i1} and H'_{i2}

Table 3
Expected process time on machines

I	H_{i1}	H_{i2}	Wi
1	9	27	3
2	20	25	5
3	26	22	2
4	2	4	1
5	10	5	4

Table 4
Weighted flow shop times

I	H'_{i1}	H'_{i2}
1	2	9
2	3	5
3	13	12
4	1	4
5	2.5	2.25

According to step 3 of the research procedure, the sequence s1 where the elements of this sequence are {4, 1, 2, 3, 5} is the optimal one that results in the least amount of time elapsed. As presented below, Table 5 represents the inflow and outflow based on **Step 4**, for schedule s1 to provide a comprehensive overview.

Table 5
Table for flow in and out of string **s1**

i	H_1	H_2
4	0 – 2	2 – 6
1	2 – 11	11 – 38
2	11 – 31	38 – 63
3	31 – 57	63 – 85
5	57 – 67	85 – 90

Thus, total elapsed time $C_{max} = 90$

As per **Step-5;** $l_2 = 90 – 83 = 7$

According to Step 6 of the research methodology, an IN-OUT table should be created to address the revised scheduling problem, as outlined in Table 6.

Table 6
Table of flow in-out for route h1→ h2 with zero idle time

Jobs	Machine H_1	Machine H_2
	Inflow- Outflow	Inflow- Outflow
4	0 – 2	7 – 11
1	2 – 11	11 – 38
2	11 – 31	38 – 63
3	31 – 57	63 – 85
5	57 – 67	85 – 90

As per **Step-7;** $u_1(s_1) = 67$

$u_2(s_1) = 90 - 7 = 83$

As per **Step-8;** $r(s_1) = u_1(s_1) * c_1 + u_2(s_1) * c_2 = 67 * 4 + 83 * 6 = 766$ units

For machine route $H_1 \rightarrow H_2$ of the optimum sequence s_1={4, 1, 2, 3, 5}, the aforementioned computed findings are thus documented in Table 7. Accordingly, the heuristic algorithm proposed for machine route $H_1 \rightarrow H_2$ yields the lowest possible rental cost and utilization time for the optimal solution s_1, as shown in Table 7.

Table 7
Evaluation of results in comparison

Machine Path $H_1 \rightarrow H_2$	Rental Costs	Utilization Time of H_2
Johnson Algorithm	796 units	88 units
Proposed Algorithm	766 units	83 units

7. Computational analysis and results

In order to analyze the suggested heuristic approach, an arbitrary number of samples for multiple groups each of which has various number of jobs are taken. A total of eight groups, each consisting of job sizes 5, 10, 20, 40, 55, 60, and 80 are created. Each group was then subjected to observation under five distinct tribulations, which were randomly generated. A comparison is made between the mean of overall rental cost in the proposed algorithm and the current make-span techniques of Palmer (1985), Johnson (1954) and NEH (1983). The results are presented in Table 8. and graph was plotted, as shown in Fig. 1, to illustrate the comparison. The findings indicate that, when compared to the remaining curves, the curve associated with the suggested approach has a lower trajectory. Notably, Palmer's algorithm demonstrates a significantly elevated curve compared to other existing approaches. Furthermore, the curve of NEH (1983) is closer than others to the proposed algorithm's curve.

Table 8
Computational experiments for total rental cost of machines

Job Size (n)	Johnson	Palmer	NEH	Proposed Algorithm
5	181.15	186.13	170.95	168.25
10	824.65	843.48	769.67	754.77
20	3663.90	3748.40	3323.13	3270.68
40	14894.48	15177.85	13399.88	13295.23
55	28534.03	29029.67	25515.53	25321.40
60	33700.85	34318.65	30345.05	30155.90
80	60686.47	61928.07	54451.72	54161.63

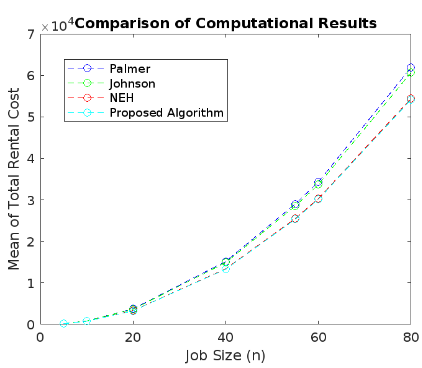

Fig. 1. Comparison of Computational Results

Table 9
Average error percentage

n	Percentage Error Mean in Johnson algorithm	Percentage Error Mean in Palmer algorithm	Percentage Error Mean in NEH algorithm
5	7.73	10.73	1.61
10	9.28	11.78	1.97
20	12.05	14.62	1.61
40	12.07	14.20	0.79
55	12.71	14.66	0.77
60	11.77	13.82	0.63
80	12.04	14.34	0.54

Moreover, to assess the quality of the suggested algorithm, calculation of error percentage for each problem follows a specific formula, denoted as E_{rr}. This formula is expressed as:

$$[(R_\delta - R_\theta)/\, R_\theta] \times 100$$

In this case, R_δ represents the overall rental cost of all currently available algorithms, while R_θ represents overall rental cost associated with same job determined when utilizing the new algorithm and results are plotted in the graph below, which is depicted in Fig. 2.

Table 10
Percentage error mean on average

Algorithm	Percentage Error Mean on Average
Johnson	11.14
Palmer	13.55
NEH	1.16

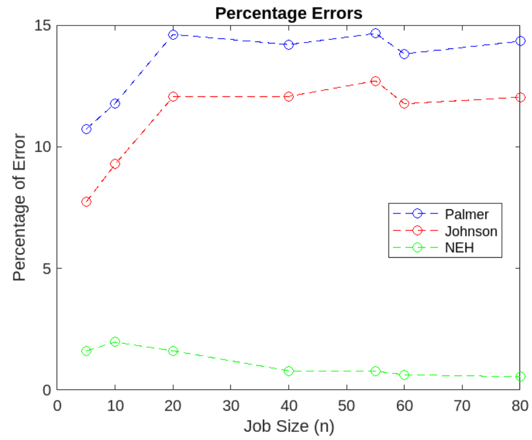

Fig. 2. Percentage Error Mean on Average

8. Conclusion

In this paper, the proposed heuristic algorithm is provided an optimal result to no-idle two stage flow shop scheduling problem while simultaneously optimizing the rental cost. The algorithm considers multiple aspects, including processing time, job weightage and separated setup times. In the present investigation, our primary objective was to attain the desired outcome across various job sizes. Earlier the researchers encompassed small-sized jobs, where the range of n was limited to (1≤n≤6) due to the complexity of computation. But we extended our efforts to encompass medium-sized jobs, with n falling within the range of 7≤n≤30. Furthermore, we sought to accomplish our goal for large-sized jobs, where the value of n ranged from 31 to 80. In this study, a series of computational tests were successfully carried out. The results of these experiments indicate that the developed heuristic algorithm surpasses the previously presented heuristics proposed by Palmer (1985), Johnso n(1954) and NEH (1983). Furthermore, this work may also be expanded by taking into account numerous aspects such as job blocking breakdown effect, transportation time etc. More time can be spent on the research by using trapezoidal fuzzy numbers to represent machine processing time .

Acknowledgment

It is with tremendous appreciation that the authors acknowledge the Chief Editor and the esteemed the members of the review panel for their diligent scrutiny of the manuscript and insightful recommendations.

References

Adiri, I., & Pohoryles, D. (1982). Flow shop no-idle or no-wait scheduling to minimize the sum of completion times. *Naval Research Logistics*, *29*(3), 495–504.

Allahverdi, A., Gupta, J. N. D., & Aldowaisan, T. (1999). A review of scheduling research involving setup considerations. *Omega, International Journal Management Science*, *27*(2), 219–239. https://doi.org/10.1016/S0305-0483(98)00042-5

Baraz, D., & Mosheiov, G. (2008). A note on a greedy heuristic for flow-shop makespan minimization with no machine idle-time. *European Journal of Operational Research*, *184*, 810–813.

Campbell, H. G., Dudek, R. A., & Smith, M. L. (1970). A heuristic algorithm for the n job, m machine sequencing problem. *Management Science*, *16*(10), B630–B637.

Chen, J., Wang, L., & Peng, Z. (2019). A collaborative optimization algorithm for energy-efficient multi-objective distributed no-idle flow-shop scheduling. *Swarm and Evolutionary Computation*, *50*, 100557. https://doi.org/10.1016/j.swevo.2019.100557

Deng, G., & Gu, X. (2012). A hybrid discrete differential evolution algorithm for the no-idle permutation flow shop scheduling problem with makespan criterion. *Computers and Operations Research*, *39*(9), 2152–2160.

Goncharov, Y., & Sevastyanov, S. (2009). The flow shop problem with noidle constraints: A review and approximation. *European Journal of Operational Research*, *196*(2), 450–456.

Gupta, D., Goel, R., & Kaur, H. (2021). Optimizing rental cost with no idle constraints in two machines with weightage. *Materials Today: Proceedings*. https://doi.org/10.1016/j.matpr.2021.01.090

Gupta D., Shashi B., S. S. (2012). To Minimize The Rental Cost For 3- Stage Specially Structured Flow Shop Scheduling with Job Weightage. *International Journal of Engineering Research and Applications (IJERA)*, *2*(3), 912–916.

Ignall, E, S. L. (1965). Application of the branch and bound technique to some flow-shop scheduling problems. *Operations Research*, *13*(3), 400–412.

Jackson, J. R. (1956). An extension of Johnson's results on job IDT scheduling. *Naval Research Logistics Quarterly*, *3*(3), 201–203.

Johnson, S.M. (1954). Optimal two-and three-stage production schedules with setup times included. *Naval Research Logistics (NRL)*, *1*(1), 61–68.

KALCZYNSKI, P. J., & KAMBUROWSKI, J. (2005). A heuristic for minimizing the makespan in no-idle permutation flow shops. *Computers and Industrial Engineering*, *49*(1), 146–154.

Kim, S. C., & Bobrowski, P. M. (1994). Impact of Sequence-Dependent Setup Time on Job Shop Scheduling Performance. *International Journal of Production Research*, *32*(7), 1503–1520.

Kumari, S., Khurana, P., & Singla, S. (2021). RAP via constraint optimization genetic algorithm. *Life Cycle Reliability and Safety Engineering*, *10*(4), 341–345. https://doi.org/10.1007/s41872-021-00173-0

Kumari, S., Khurana, P., & Singla, S. (2022). Behavior and profit analysis of a thresher plant under steady state. *International Journal of System Assurance Engineering and Management*, *13*(1), 166–171. https://doi.org/10.1007/s13198-021-01183-y

Maggu, P. L., & Das, G. (1982). Weighted flow shop scheduling Problem. In *Elements of Advanced Production Scheduling* (1985th ed., pp. 105–111). United Publishers and Periodical distribution.

Miyazaki, S., Nishiyama, N. (1980). Analysis for minimizing weighted mean Minimizing rental cost in two stage flow shop, the processing time associated with probabilities including job block. *Reflections de ERA*, *1*(2), 107–120.

Nawaz, M., Enscore Jr., E. E., & Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow shop sequencing problem. *The International Journal of Management Science, 11*(1), 91–95.

PALMER, D. S. (1985). Sequencing jobs through a multi stage process in the minimum total time-A quick method for obtaining a near optimum. *Operations Research*, *16*, 101–107.

Pan, Q. K., & Wang, L. (2008a). A novel differential evolution algorithm for no-idle permutation flow-shop scheduling problems. *European Journal of Industrial Engineering*, *2*(3), 279–297.

Pan, Q. K., & Wang, L. (2008b). No-idle permutation flow shop scheduling based on a hybrid discrete particle swarm optimization algorithm. *International Journal of Advanced Manufacturing Technonlogy.*, *39*(7–8), 796–807.

Rad, S. F., Ruiz, R., & Boroojerdian, N. (2009). New high performing heuristics for minimizing makespan in permutation flowshops. *OMEGA, the International Journal of Management Science*, *37*, 331–345.

RUIZ, R., & STÜTZLE, T. (2007). A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, *177*(3), 2033–2049.

Ruiz, R., Vallada, E., & Fernández-Martínez, C. (2009). Scheduling in Flowshops with No-Idle Machines. In U. K. Chakraborty (Ed.), *Computational intelligence in flow shop and job shop scheduling* (Vol. 230, pp. 21–51). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-02836-6_2

Shao, W., Pi, D., & Shao, Z. (2018). Local search methods for a distributed assembly no-idle flow shop scheduling problem. *IEEE Systems Journal*, *13*(2), 1945–1956.

Singla, S., Kaur, H., & Gupta, D. (2023). Optimization of Rental Cost of Machines in Two Stage No-Idle Scheduling with Transportation Time and Weightage of Jobs. *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*, 818–821. https://ieeexplore.ieee.org/document/10112467

Singla, S., Kaur, H., Gupta, D., & Kaur, J. (2023a). No Idle Constraint In Flow Shop Scheduling With Transportation Time, Weightage of Jobs And Job Block Criteria. *2023 IEEE 2nd International Conference on Industrial Electronics: Developments & Applications (ICIDeA)*, 450–454. https://doi.org/10.1109/ICIDeA59866.2023.10295180

Singla, S., Kaur, H., Gupta, D., & Kaur, J. (2023b). Two Stage No Idle Flow Shop Scheduling To Minimize Rental Cost Including Transportation Time. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1–4. https://doi.org/10.1109/ICCCNT56998.2023.10306808

Taillard, E. (1990). Some efficient heuristic methods for the flow shop sequencing problem. *European Journal of Operational Research* , *47*, 67–74.

Taillard, E. (1993). Benchmarks for basic scheduling problems. *European Journal of Operational Research*, *64*, 278–285.

Yazdani, M., & Naderi, B. (2016). Modeling and scheduling no-idle hybrid flow shop problems. *Journal of Optimization in Industrial Engineering*, *10*(21), 59–66.

Yoshida, T., & Hitomi, K. (1979). Optimal two-stage production scheduling with setup times separated. *AIIE Transactions*, *11*(3), 261–263.

Zhang, W., Hou, W., Li, C., Yang, W., & Gen, M. (2021). Multidirection Update-Based Multiobjective Particle Swarm Optimization for Mixed No-Idle Flow-Shop Scheduling Problem. *Complex System Modeling and Simulation*, *1*(3), 176–197. https://doi.org/10.23919/CSMS.2021.0017

Zhou, Y., Chen, H., & Zhou, G. (2014). Invasive weed optimization algorithm for optimization no-idle flow shop scheduling problem. *Neurocomputing*, *137*, 285–292.