Dhaka Univ. J. Sci. 74(2): 341–352, 2026 (July) ISSN 1022-2502, E-ISSN 2408-8528
DOI: https://doi.org/10.3329/dujs.v74i2.86872

# A Proposed Heuristic Algorithm for $n$-job $m$-machine Job Sequencing Problem

**Esrat Jahan Meem, Md. Asadujjaman* and Isnat Jahan Owishi**

*Department of Mathematics, University of Dhaka, Dhaka-1000, Bangladesh*

(Received: 4 January 2026; Accepted: 8 April 2026)

**Abstract**

This paper presents an in-depth analysis of job sequencing in flow-shop scheduling problems, with a focus on exploring alternative mean-based techniques alongside the classical Johnson's method. While Johnson's method is widely recognized for determining optimal job sequences in two and three machine problems, also examines the impact of applying different mean-based approaches such as the Arithmetic Mean, Harmonic Mean, Contra-harmonic Mean, and Quadratic Mean on sequencing decisions and performance outcomes. The comparative analysis is carried out using identical problem instances to ensure consistency in evaluation. Results indicate that, in certain cases, mean-based methods produce sequences identical to Johnson's method, thereby preserving the optimal elapsed time, while in other cases, variations in job sequences occur without altering the total elapsed time, thus maintaining optimality. However, differences in machine idle times are observed, suggesting potential efficiency improvements in specific scenarios. The findings highlight that mean based methods can serve as viable alternatives to Johnson's method, offering flexibility in sequencing decisionswhile retaining solution quality. This work contributes to the understanding of heuristic adaptations in classical scheduling problems and provides insights for practitioners seeking efficient scheduling strategies in manufacturing and service operations.

**Keywords:** Job Sequencing, Single Machine Sequencing Problem, Meantechnique, Contra-harmonic Mean, Johnson's method, Optimal Sequencing.

## I. Introduction

In mathematics, Job Sequencing is a process of scheduling a set of jobs in a manner which gives the maximum outcome, reduces the machine idle time and total elapsed time that is keep the machines working and give best sequence of jobs. In the field of operations research and production management, job sequencing refers to the process of determining the most efficient order in which a set of jobs should be performed. This problem becomes particularly important when multiple jobs compete for limited resources, such as machines or workers, and the goal is to optimize performance measures like total completion time, lateness, or idle time. The primary objective in multi-machine job sequencing is often to minimize the total completion time (make span), reduce machine idle time, or improve overall system efficiency[1,2,3]. Depending on the nature of the machine arrangement and job flow, different solution approaches are used. For instance, Johnson's Algorithm provides an efficient method for sequencing jobs through two or three machines in a specific order. In more complex settings, techniques such as Branch and Bound, Dynamic Programming, and various meta heuristic algorithms are applied to find near-optimal or optimal schedules. But sometimes classic algorithms become too complex for large-scale, multi-machine problems and existing solutions become computationally prohibitive as the problem size scales[4,5,6]. So there is a There is a need for a practical, efficient, and accessible approach. Our proposed method is not only simple, computationally efficient, and reliable but also provides results which are consistent with established algorithms, evenas the problem size and complexity increase. The rest of the paper is organized as follows. In section 2 we discuss on some basic definitions related to Job sequencing problems. In section 3 we discuss an existing method and existing algorithm for Job sequencing problems. In section 4 we discuss our proposed algorithm for Job sequencing problems. In section 5 we illustrate the solution procedure with a number of numerical examples and compare the solutions with existing method. Finally, we draw a conclusion in section 6.

## II. Preliminaries

In this section, we briefly discuss definitions that are used in job sequencing problems. There are two type sequencing problems single machine sequencing problem and multi-machine sequencing problems[8,9].

Single Machine Sequencing Problem: A pure sequencing problem is a type of scheduling issue where the order of jobs alone determines the schedule. The simplest case involves a single machine with deterministic processing times. Despite its simplicity, this model is crucial as it helps explore various performance measures and solution techniques in a manageable way. It often appears as a component in more complex scheduling systems, such as bottlenecks in multi-stage operations, and can sometimes be solved separately to improve the overall schedule.

* Author for correspondence. e-mail: asad132@du.ac.bd

Besides being limited to a single machine, the basic scheduling problem is defined by the following circumstances:

$C1$. At time zero, there are $n$ jobs that can be processed, each of which requires a single operation.

$C2$. The machine can handle only one job at a time.

$C3$. Setup times for jobs do not depend on the order of the jobs and are included within the processing times.

$C4$. All job details are known beforehand and are deterministic.

$C5$. The machine is continuously available without any breakdowns.

$C6$. The machine is never left idle as long as there are jobs waiting to be processed.

$C7$. Once a job starts, it continues without any interruptions.

Given these assumptions, every possible sequence of jobs corresponds exactly to one permutation of the job indices $1, 2, \dots, n$.Therefore, the sum of the number of the unique solutions for this basic single-machine scheduling problem is $n!$, representing all possible sequences of $n$ jobs[11,12].

When considering job characteristics in the single-machine model, this is important to differentiate between data that is available beforehand and data that becomes known only after scheduling decisions are made. Information that is already known is treatedas input to this process and is typically represented using lowercase notation. In this context, three fundamental pieces of information are commonly used to describe jobs in the single-machine setting:[7,10,13].

**Processing time ($p_j$):** The amount of processing required by job $j$.

**Release date ($r_j$):** The time at which job $j$ is available for processing.

**Due date ($d_j$):** The time at which the processing of job $j$ is due to be completed.

Under condition C3 the processing time $p_j$ generally includes both direct processing time and facility setup time. The release date can be thought of as an arrival time, the time when job $j$ appears at the processing facility and in the basic model, the assumption in condition C1 is that$r_j = 0$ for all jobs[14,15].

Due dates might not be relevant for some problems, but they are often an important factor in scheduling tasks. The basic model can help clarify goals related to meeting due dates. Data produced by scheduling decisions is considered output from the scheduling process, and this type of information is typically represented using capital letters. These scheduling decisions provide the key data needed to assess the quality of schedules.

**Completion time ($C_j$):** When job $j'$s processing is complete. Schedules are typically evaluated quantitatively using job completion times as a function. There are two significant quantities:

**Flow time ($F_j$):**The total time that job $j$ remains in the system is given by: $F_j = C_j - r_j$. Where,$C_j$ is the completion time of the job and $r_j$ is its release time.

**Lateness ($L_j$):** The length of time that job j finishes later than its due date: $L_j = C_j - d_j$.

Flow time and lateness represent two different aspects of service performance. Flow time (or turnaround time) is the total time a job spends in the system from arrival to completion reflecting how quickly the system responds. Lateness measures how closely a job's completion aligns with its due date; it can be negative (early completion) or positive (late completion). While early completion often carries no benefit, positive lateness is usually penalized in many scheduling environments.

Hence, it is often useful to use a measure that accounts only for delays beyond the due date.

**Tardiness ($T_j$):** The lateness of job $j$ when it misses it's due date, or zero if it is on time. Otherwise $T_j = max\{0, L_j\}$. Schedules are typically assessed using aggregate measures that incorporate data from all jobs, producing single-value performance indicators. These performance measures are generally based on the set of completion times in a schedule. For instance, if $n$ jobs are to be scheduled, possible aggregate performance measures could include the following:

**Total flow time:**$F = \sum_{j=1}^{n} F_j$

**Total tardiness:**$T = \sum_{j=1}^{n} T_j$

**Maximum flow time:**$F_{max} = \max_{1 \le j \le n} \{F_j\}$

**Maximum tardiness:**$T_{max} = \max_{1 \le j \le n} \{T_j\}$

The number of jobs that are late or the total amount of their unit penalties:

$$U = \sum_{j=1}^{n} \delta(T_j),$$

Where, $\delta(x) = 1\ if x > 0$ and $\delta(x) = 0$ otherwise

**Maximum completion time:**$C_{max} = \max_{1 \le j \le n} \{C_j\}$

## III. Existing Method for Job Sequencing Problems

In this section, we discuss the existing method (Johnson's Method) and existing algorithm for Job Sequencing problems **[12].**

*Processing n Jobs through Two Machines*

This sequencing problem is completely described as follows:

(i) Only two machines are involved $A$ and $B$

(ii) Each job is processed in the order $AB$ and

(iii) The actual or expected processing times $A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_n$ are known and represented by a take of the type shown below.

**Machine time for n jobs and two machines**

| Job $i$ | $A$ | $B$ |
|---|---|---|
| 1 | $A_1$ | $B_1$ |
| 2 | $A_2$ | $B_2$ |
| 3 | $A_3$ | $B_3$ |
| . | . | . |
| . | . | . |
| $i$ | $A_i$ | $B_i$ |
| . | . | . |
| . | . | . |
| $n$ | $A_n$ | $B_n$ |

The problem is to determine the sequence (order) of jobs which minimizes T, the total elapsed time from the start of a first job to the completion of last job. The solution procedure of the following steps:

**Step 1:** Select the least processing timer occurring in the list $A_1, A_2, \dots, A_n$ and $B_1, B_2, \dots B_n$.

**Step 2:** If the shortest processing time is for machine $A$ then put it at the start of the sequence. If it's for machine $B$, put it at the end.

**Step 3:** There may be three options if there is a tie in determining the minimum processing time.

(a) If machine $A$ is the only one with an equal minimum value, choose the job in $B$ with the smallest processing time to be placed first in the job sequence.

(b) Choose the job with the shortest processing time in machine $A$ to be placed last in the job sequence if equal minimum values only occur for machine $B$.

(c) If there are equal minimum values for both machines, then place the job in machine $A$ first and the one in machine $B$ last.

**Step 4:** If every job has been sequenced, proceed to the next step; if not, repeat Steps 1 through 3. Delete the jobs that have already been sequenced.

**Step 5:** Calculates the idle time on machines A and B as well as the total amount of time that has passed (total elapsed time).

*Processing n Jobs through Three Machines*

Transform the problem into an equivalent two-machine problem, and then solve it using Johnson's algorithm

**Step 1:** Determine the maximum processing time for the second machine and the minimum processing time for the first and last machines.

$i.e$, find Min$(A_i, C_i)[i = 1,2,3, \dots, n]$

and Max$(B_i)$

**Step 2:** Check the following inequality $(Min(A_i) \geq Max(B_i)$ or $(Min(C_i) \geq Max(B_i))$

**Step 3:**This approach cannot be used if none of the step 2 inequalities are met.

**Step 4:** We define two machines, $G$ and $H$, such that the processing times on $G$ and $H$ are given by, provided that at least one of the inequalities in step 2 is satisfied.

$$G_i = A_i + B_i$$

$$H_i = B_i + C_i (i = 1,2,3, \dots, n)$$

**Step 5:** We use the two-machine algorithm to determine the optimal sequence for the converted machines $G$ and $H$**[5, 6, 12].**

*Processing $n$ Jobs through $m$ Machines*

Transform the given problem into two machine problem and then solve using **Johnson's algorithm**

**Step 1:** Find minimum $M_i$ and minimum $M_k$ and maximum of each $M_2$ ,$M_3$ ,…,$M_{k-1}$ ($M_k$ is last machine)

**Step 2:** Check whether

$$Min(M_i) \geq Max(M_j) \text{ or}$$

$$Min(M_k) \geq Max(M_j) (j = 2,3, \dots, k-1)$$

**Step 3:** The method fails if the inequalities in step 2 are not met. If they are, move on to the next step.

**Step 4:** Along with step 2, if $M_2 + M_3 + \dots + M_{k-1} = C$ where $C$ is positive constant for all $i = 1,2, \dots, n$.

The optimal sequence algorithm is then used to determine the best order for $n$ jobs,where two machines are $M_1$ and $M_k$.

**Step 5:** If the condition $M_2 + M_3 + \dots + M_{k-1} \neq C$, we define two machine$G$ and $H$ such that

$$G = M_1 + M_2 + \dots + M_{k-1}$$

$$H = M_2 + M_3 + \dots + M_k$$

Determine the optimal sequence using two machine algorithm or Johnson's algorithm.

## IV. Proposed Algorithm for n-Job m-Machine Job Sequencing Problem

We introduce a new approach, termed the Mean Processing Time Method, to determine the optimal job sequence in job sequencing problems. Consider $n$ jobs to be processed on $m$

machines in the same sequence, where the processing time of job $i$ on machine $j$ is $t_{i,j}$ $(i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m)$. The objective is to determine the order of jobs such that the total elapsed time (make span) is minimized.For each job $i$, calculate the mean processing time by arithmetic mean as :$\overline{T_i} = \frac{\sum_{j=1}^{m} t_{ij}}{m}$

After computing the mean processing times for all jobs, arrange them in descending order of $\overline{T_i}$. Then select the job corresponding to the **minimum mean value**. Then, check the processing times of that job on machines $M_1$ and $M_2$.

- If the processing time is minimum on machine $M_1$ , the job is placed in the **first position** of the sequence.
- If the processing time is minimum on machine $M_2$, the job is placed in the **last position** of the sequence.

This procedure is repeated until all jobs are assigned a position, thereby generating the optimal sequence.

*Algorithm: Optimal Sequencing for the n-Job, 2-Machine Problem (Mean Value Approach)*

**Step 1: Start**

Begin the algorithm.

**Step 2: Input and Initialization**

Read the number of jobs $n$, and the processing times for each job $j = 1,2,\ldots,n$ on machine $M_1$ $(T_{1j})$ and machine $M_2$ $(T_{2j})$. Initialize an empty optimal sequence array.

**Step 3: Calculate Mean Values**

For each unassigned job $j$, calculate the mean processing time $\mu_j$, as the average of its processing times on machine $M_1$ and $M_2$.

$$\mu_j = \frac{T_{1j} + T_{2j}}{2}$$

**Step 4: Find the Minimum Mean Value**

Identify the job $k$ with the minimum mean processing time among all unassigned jobs.

$$\mu_{min} = \min_{j \in Unassigned\ jobs} (\mu_j)$$

**Step 5: Apply Sequencing Rule**

Based on the job with the minimum mean value, apply the following rule:

- **Case 1: Minimum on Machine $M_1$:** If the minimum mean value corresponds to a job $k$ where $T_{1k} \le T_{2k}$, then place job $k$ in the first available position of the optimal sequence.
- **Case 2: Minimum on Machine $M_2$:** If the minimum mean value corresponds to a job $k$ where $T_{2k} < T_{1k}$, then place job $k$ in the last available position of the optimal sequence.

**Step 6: Handle Ties**

If a tie occurs in selecting the minimum mean processing time, apply the following tie-breaking rules:

- **Tie across machines:** If $\min(\mu_j) = \mu_k = \mu_r$, and the tie-breaking condition is $\frac{T_{1k}+T_{2k}}{2} = \frac{T_{1r}+T_{2r}}{2}$ where $T_{1k}$ is the minimum for job $k$ and $T_{2r}$ is the minimum for job $r$, then process job $k$ first and job $r$ last.
- **Tie on Machine $M_1$ only:** If multiple unassigned jobs have the same minimum mean value and their minimum processing time is on $M_1$, select the job with the smallest job subscript first.
- **Tie on Machine $M_2$ only:** If multiple unassigned jobs have the same minimum mean value and their minimum processing time is on $M_2$, select the job with the largest job subscript last.

**Step 7: Update and Repeat**

Remove the sequenced job from the list of unassigned jobs. Repeat steps 3 through 6 until all n jobs have been assigned a position in the optimal sequence.

**Step 8: Calculate Performance Metrics**

Once the optimal sequence is determined, calculate the following:

- **Total Elapsed Time:** The time required to complete all jobs.
- **Idle Time:** The idle time for both machine $M_1$ and machine $M_2$.

**Step 9: End**

The algorithm concludes.

*Problems Involving Processing $n$Jobs through 3 Machines*

Let there be n-jobs each of which is to be processed through 3-machines, $M_1, M_2$ and $M_3$ in the order $M_1M_2M_3$. Let $T_{ij}(i = 1,2,3\ ;\ j = 1,2,\ldots\ldots n)$ be the required time for processing $i$ th job on $j$th machine. The table shows the time duration as follows:

| Jobs/Machines | $J_1$ | $J_2$ | …… | $J_n$ |
|---|---|---|---|---|
| $M_1$ | $T_{11}$ | $T_{12}$ | …… | $T_{1n}$ |
| $M_2$ | $T_{21}$ | $T_{22}$ | …… | $T_{2n}$ |
| $M_3$ | $T_{31}$ | $T_{32}$ | …… | $T_{3n}$ |

*Mean Processing Time Method for Processing n-Jobs through 3 Machines*

Transform the $\boldsymbol{n}$-job, 3-machine problem into an equivalent $n$-job, 2-machine problem by verifying either of the following conditions:

1. $Minimum\ of\ M_1 \geq Maximum\ of\ M_2$
2. $Minimum\ of\ M_3 \geq Maximum\ of\ M_2$

Or both satisfies.

**Algorithm**

1. Check the processing times of all jobs on the three machines. If at least one of the above criteria is satisfied, proceed to Step 2; otherwise, the algorithm cannot be applied.
2. Define two machines **G** and **H** with processing times:

$$T_{Gj} = T_{1j} + T_{2j}, \quad j = 1,2,\ldots\ldots,n$$

That is, the machine $G$ is defined in such a way that it is processing time is the sum of the processing times on machines $M_1$ and $M_2$.

$$T_{Hj} = T_{2j} + T_{3j}, \quad j = 1,2,\ldots\ldots,n$$

Similarly, the machine $H$ is defined in such a way that it is processing time is the sum of the processing times on machines $M_2$and $M_3$.

3. Solve the resulting $\boldsymbol{n}$-job, 2-machine sequencing problem using the prescribed machine order $\boldsymbol{G} \rightarrow H$ as discussed earlier.
4. Apply the method for the reduced $\boldsymbol{n}$-job, 2-machine problem to obtain the optimal job sequence for the original problem.

**Processing of $n$-Jobs through $m$-Machines**

| Jobs/Machines | $J_1$ | $J_2$ | …………… | $J_n$ |
|---|---|---|---|---|
| $M_1$ | $T_{11}$ | $T_{12}$ | …………… | $T_{1n}$ |
| $M_2$ | $T_{21}$ | $T_{22}$ | …………… | $T_{2n}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $M_n$ | $T_{m1}$ | $T_{m2}$ | …………… | $T_{mn}$ |

Transform the n-jobs 3-machine problems into n-jobs 2 machine problems by verifying the inequalities.

$$Minimum\ of\ M_1 \geq Maximum\ of\ (M_2, M_3, \ldots\ldots\ldots, M_{m-1})$$

$$Minimum\ of\ M_m \geq Maximum\ of\ (M_2, M_3, \ldots\ldots\ldots, M_{m-1})$$

or if both conditions hold, use the procedure discussed earlier. Then, apply the same method to the reduced $\boldsymbol{n}$-job, 2-machine sequencing problem obtained from these inequalities to determine the optimal job sequence[19,20].

To strengthen the analysis and explore the impact of different averaging approaches, this study incorporates additional mean calculation techniques alongside the Arithmetic Mean. Specifically, the Contra harmonic Mean, Quadratic Mean, and Harmonic Mean techniques have been applied to the same dataset. Each of these methods emphasizes data values differently, thereby offering distinct perspectives on the central tendency. By comparing the results obtained from these various mean techniques, the research aims to identify any significant variations in outcomes and to assess how the choice of averaging method may influence the interpretation of the results. This comparative analysis provides a more robust and comprehensive understanding of the dataset.

**New Contra harmonic Mean Technique:**

$$CHM_{av} = \frac{\left(\frac{M_1^2 + M_2^2}{2}\right)}{\left(\frac{M_1 + M_2}{2}\right)}$$

**Quadratic Mean Technique:**

$$QM_{av} = \sqrt{\frac{1}{2}(M_1^2 + M_2^2)}$$

**Harmonic Mean Technique:**

$$HM_{av} = \frac{2}{\frac{1}{M_1} + \frac{1}{M_2}}$$

*Problems Involving Processing n-Jobs through 2 Machines*

**Example:** Determine the optimal sequence of jobs which minimizes the total elapsed time required to complete the following jobs on machines $M_1$ and $M_2$ in the order $M_1 M_2$.

| Jobs/Machine | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| $M_1$ | 3 | 12 | 5 | 2 | 9 | 11 |
| $M_2$ | 8 | 10 | 9 | 6 | 3 | 1 |

**Solution:** Firstly, solving this problem using **Johnson's method** then we will apply our proposed method.

Let us make a sequence table.

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |

By examining the columns, we find the smallest value. It is threading time of 1 minute for job 6 in second column. Thus we schedule job 6 last as shown.

The next smallest value is turning time of 2 minutes for job 4 in first column Thus we schedule job 4 first as shown.

The machine used for turning (first column) goes from the left to right .Whenever we find a minimum value (time) in the first column for a job, we assign that job to the left most vacant box possible in our sequencing .The machine used for threading (second column ) goes from right to left . Whenever we find a minimum value (time) in the second column for a job, we assign that job to the right most vacant box possible in our sequencing.

There are two equal minimal values turning time of 3 minutes for job 1 in first column and threading time of 3 minutes for job 5 in second column. According to the rules, job 1 is scheduled next to job 4 and job 5 next to job 6 as shown.

The smallest value is turning time of 5 minutes for job 3 in first column. Therefore, we schedule job 3, next to job 1 and we get the optimal sequence as shown.

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_1$ | $J_3$ | $J_2$ | $J_5$ | $J_6$ |

Now, we can calculate the elapsed time corresponding to the optimal sequence, using the individual processing times given in the problem. The details are shown in table.

Total elapsed time

| Machine | Turning | | Idle Time For turning | Threading | | Idle Time for Threading |
|---|---|---|---|---|---|---|
| | Time In | Time Out | | Time In | Time Out | |
| 4 | 0 | 2 | – | 2 | 8 | 2 |
| 1 | 2 | 5 | – | 8 | 16 | – |
| 3 | 5 | 10 | – | 16 | 25 | – |
| 2 | 10 | 22 | – | 25 | 35 | – |
| 5 | 22 | 31 | – | 35 | 38 | – |
| 6 | 31 | 42 | 1 | 42 | 43 | 4 |

Total elapsed time is 43 minutes

Idle time for Machine 1 is 1 minute

Idle time for Machine 2 is $(2 + 4)$ mins $= 6$ mins

The arithmetic mean$\left(\frac{M_1+M_2}{2}\right)$ value for each job of the given table on machines $M_1$and $M_2$ is:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 5.5 | 11 | 7 | 4 | 6 | 6 |

The required job sequence for the given table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_1$ | $J_3$ | $J_2$ | $J_5$ | $J_6$ |

Which is identical to the sequence we have using Johnson's method. So total elapsed time and idle time for both machine are same.

The **contra-harmonic mean** ($\boldsymbol{CHM_{av}} = \frac{\left(\frac{M_1^2+M_2^2}{2}\right)}{\left(\frac{M_1+M_2}{2}\right)}$) value of the given table for each job on machines $M_1$and $M_2$ is:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 6.64 | 11.09 | 7.57 | 5 | 7.5 | 10.17 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_1$ | $J_3$ | $J_2$ | $J_6$ | $J_5$ |

Total elapsed time

| Jobs sequence /machines | Machine-$M_1$ | | Idle Time | Machine-$M_2$ | | Idle Time |
|---|---|---|---|---|---|---|
| | Time In | Time Out | | Time In | Time Out | |
| 4 | 0 | 2 | – | 2 | 8 | 2 |
| 1 | 2 | 5 | – | 8 | 16 | – |
| 3 | 5 | 10 | – | 16 | 25 | – |
| 2 | 10 | 22 | – | 25 | 35 | – |
| 6 | 22 | 33 | – | 35 | 36 | – |
| 5 | 33 | 42 | – | 42 | 45 | 6 |

Hence, the total elapsed time is 45 minutes and the idle time on machine$M_1$and $M_2$ are 3 minutes and 8 minutes respectively.

The **Quadratic mean** $\boldsymbol{QM_{av}} = \sqrt{\frac{1}{2}(\boldsymbol{M_1^2} + \boldsymbol{M_2^2})}$ of the given table for each job on machines $M_1$and $M_2$ is:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 6.04 | 11.05 | 7.28 | 4.47 | 6.71 | 7.81 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_1$ | $J_3$ | $J_2$ | $J_6$ | $J_5$ |

Which is identical to the sequence we have using **contra-harmonic mean** method. So total elapsed time and idle time for both machine are same as it.

The **Harmonic mean** ($\boldsymbol{HM_{av}} = \frac{2}{\frac{1}{M_1}+\frac{1}{M_2}}$)value of the given table for each job on machines $M_1$and $M_2$ is:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 4.36 | 10.9 | 6.43 | 3 | 4.5 | 1.83 |

The required job sequence for the given table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_1$ | $J_3$ | $J_2$ | $J_5$ | $J_6$ |

Which is identical to the sequence we have using Johnson's method. So total elapsed time and idle time for both machine are same.

So we can conclude that, in some mean techniques we have identical solution as Johnson's method but in other cases total elapsed time slightly changes but it has near optimal value. So for n-job 2-machine problem Johnson's method gives the optimal value and arithmetic and Harmonic mean method give same result as it[16,17].

*Problems Involving Processing n-Jobs through 3-Machines*

**Example:**Identify the sequence that results in the minimum total elapsed time for completing the following tasks on the machines in the given order $M_1M_2M_3$. Find also minimum total elapsed time and the idle time on the machines.

| Jobs/Machines | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| $M_1$ | 3 | 12 | 5 | 2 | 9 | 11 |
| $M_2$ | 8 | 6 | 4 | 6 | 3 | 1 |
| $M_3$ | 13 | 14 | 9 | 12 | 8 | 13 |

**Solution:** Firstly, solving this problem using **Johnson's method** then we will apply our proposed method.

$$Min(M_1) = 2, Max(M_2) = 8, Min(M_3) = 8$$

$$Min(A) \geq Max(M_2)$$

$$Or, 2 \geq 8\ (False)$$

$$Or, Min\ (M_3) \geq\ Max\ (M_2)$$

$$Or, 8 \geq 8\ (True)$$

We define two machine $G$ and $H$

Such that, $= M_1 + M_2$ , $H = M_2 + M_3$

| Job | G | H |
|---|---|---|
| 1 | 11 | 21 |
| 2 | 18 | 20 |
| 3 | 9 | 13 |
| 4 | 8 | 18 |
| 5 | 12 | 11 |
| 6 | 12 | 14 |

Optimal sequence for the job is,

| | | | | | |
|---|---|---|---|---|---|
| $J_4$ | $J_3$ | $J_1$ | $J_6$ | $J_2$ | $J_5$ |

Total Elapsed Time

| Jobs sequence /machines | Machine-$A$ | | Idle Time | Machine-$B$ | | Idle Time | Machine-$C$ | | Idle Time |
|---|---|---|---|---|---|---|---|---|---|
| | Time In | Time Out | | Time In | Time Out | | Time In | Time Out | |
| $J_4$ | 0 | 2 | – | 2 | 8 | 2 | 8 | 20 | 8 |
| $J_3$ | 2 | 7 | – | 8 | 12 | – | 20 | 29 | – |
| $J_1$ | 7 | 10 | – | 12 | 20 | – | 29 | 42 | – |
| $J_6$ | 10 | 21 | – | 21 | 22 | 1 | 42 | 55 | – |
| $J_2$ | 21 | 33 | – | 33 | 39 | 11 | 55 | 69 | – |
| $J_5$ | 33 | 42 | – | 42 | 45 | 3 | 69 | 77 | – |

Total elapsed time $= 77$ minutes

Idle time for machine $A = (77 - 42) = 35$ minutes

Idle time for machine $B = (77 - 45) + 2 + 1 + 11 + 3 = 49$ minutes

Idle time for machine $C\ = (77 - 77) + 8 = 8$ minutes

The arithmetic mean of the rows of the given table for each job on machines $M_1$ and $M_2$ is follows:

| Jobs/Machines | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| $G$ | 11 | 18 | 9 | 8 | 12 | 12 |
| $H$ | 21 | 20 | 13 | 18 | 11 | 14 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_3$ | $J_4$ | $J_6$ | $J_1$ | $J_2$ | $J_5$ |

**Total Elapsed Time**

| Jobs sequence /machines | Machine-$M_1$ | | Idle Time | Machine-$M_2$ | | Idle Time | Machine-$M_3$ | | Idle Time |
|---|---|---|---|---|---|---|---|---|---|
| | Time In | Time Out | | Time In | Time Out | | Time In | Time Out | |
| $J_3$ | 0 | 5 | – | 5 | 9 | 5 | 9 | 18 | 9 |
| $J_4$ | 5 | 7 | – | 9 | 15 | – | 18 | 30 | – |
| $J_6$ | 7 | 18 | – | 18 | 19 | 3 | 30 | 43 | – |
| $J_1$ | 18 | 21 | – | 21 | 29 | 2 | 43 | 56 | – |
| $J_2$ | 21 | 33 | – | 33 | 39 | 4 | 56 | 69 | – |
| $J_5$ | 33 | 44 | – | 44 | 47 | 5 | 69 | 77 | – |

Total elapsed time $= 77$ minutes

Idle time :

For machine $M_1 \; = \; (77-44) = 33$ minutes

For machine $M_2 \; = (77-47)+5+3+2+4+5 = 49$ minutes

For machine $M_3 \; = (77-77)+9 = 9$ minutes

The contra-harmonic mean of the rows of the given table for each job on machines $G$ and $H$ is follows:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 17.56 | 19.05 | 11.36 | 14.92 | 11.52 | 13.08 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_3$ | $J_6$ | $J_4$ | $J_1$ | $J_2$ | $J_5$ |

Total elapsed time

| Job Sequence/Machines | Machine-$M_1$ | | Idle Time | Machine-$M_2$ | | Idle Time | Machine-$M_3$ | | Idle Time |
|---|---|---|---|---|---|---|---|---|---|
| | Time In | Time Out | | Time In | Time Out | | Time In | Time Out | |
| $J_3$ | 0 | 5 | – | 5 | 9 | 5 | 9 | 18 | 9 |
| $J_6$ | 5 | 16 | – | 16 | 17 | 7 | 18 | 31 | – |
| $J_4$ | 16 | 18 | – | 18 | 24 | 1 | 31 | 43 | – |
| $J_1$ | 18 | 21 | – | 24 | 32 | – | 43 | 56 | – |
| $J_2$ | 21 | 33 | – | 33 | 39 | 1 | 56 | 70 | – |
| $J_5$ | 33 | 42 | – | 42 | 45 | 3 | 70 | 78 | – |

Total elapsed time $=78$ mins, Idle time for machine $A = 36$ mins , idle time for machine $B = \; 17$ mins, idle time for machine $C \; = \; 9$ mins

The Quadratic mean $QM_{av} = \sqrt{\frac{1}{2}(M_1^2+M_2^2)}$ of the rows of the given table for each job on machines $M_1$ and $M_2$ is follows:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 16.76 | 19.03 | 11.18 | 13.93 | 11.51 | 13.04 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_3$ | $J_6$ | $J_4$ | $J_1$ | $J_2$ | $J_5$ |

Which is identical to the sequence we have using contra-harmonic mean method. So total elapsed time and idle time for both machine are same as it.

The Harmonic mean of the rows of the given table for each job on Machines $M_1$ and $M_2$ is follows:

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ |
|---|---|---|---|---|---|---|
| Mean | 14.44 | 18.95 | 10.64 | 11.08 | 11.48 | 12.93 |

The job sequence for the above table is:

| | | | | | |
|---|---|---|---|---|---|
| $J_3$ | $J_4$ | $J_6$ | $J_1$ | $J_2$ | $J_5$ |

Which is identical to the sequence we have using Arithmetic mean method. So total elapsed time and idle time for both machine are same.

In this case we can say all the techniques give more or less similar total elapsed time. Here we can see by using Quadratic and Contra-Harmonic mean we can reduce the idle time for machine-2 and machine-1 which is very effective in real life scenarios. So for n-job 3-machine problem our proposed technique performs better.

*Problems Involving Processing n-Jobs through m-Machines*

**Example:** Four Jobs are to be processed on each of the 5 machines. Find the total minimum elapsed time if no passing of jobs is permitted. Also find the idle time for each machine.

| Jobs/Machines | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| *A* | 7 | 6 | 5 | 8 |
| *B* | 5 | 6 | 4 | 3 |
| *C* | 2 | 4 | 5 | 3 |
| *D* | 3 | 5 | 6 | 2 |
| *E* | 9 | 10 | 8 | 6 |

**Solution:** Since the given problem involves sequencing on five machines, it is first transformed into an equivalent two-machine problem.

$$Min(A) = 5, Min\ (E) = 6, Max\ (Max(B), Max(C), Max(D)) = Max(6,5,6) = 6$$

$$Min(E) \geq Max(6,5,6)(True)$$

Since, $B_1 + C_1 + D_1 = 10$

$$B_2 + C_2 + D_2 = 15$$

$$\therefore B_i + C_i + D_i \neq C$$

Let $G$ and $H$ be two machines such that,

$$G = A + B + C + D$$
$$H = B + C + D + E$$

| Jobs | G | H |
|---|---|---|
| 1 | 17 | 19 |
| 2 | 21 | 25 |
| 3 | 20 | 23 |
| 4 | 16 | 14 |

Optimal Sequence is

| | | | |
|---|---|---|---|
| 1 | 3 | 2 | 4 |

**Total elapsed time**

| Job | Machine A | | Idle Time | Machine B | | Idle Time | Machine C | | Idle Time | Machine D | | Idle Time | Machine E | | Idle Time |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | In | Out | | In | Out | | In | Out | | In | Out | | In | Out | |
| 1 | 0 | 7 | – | 7 | 12 | 7 | 12 | 14 | 12 | 14 | 17 | 14 | 17 | 26 | 18 |
| 3 | 7 | 12 | – | 12 | 16 | – | 16 | 21 | 2 | 21 | 27 | 4 | 27 | 35 | 1 |
| 2 | 12 | 18 | – | 18 | 24 | 2 | 24 | 28 | 3 | 28 | 33 | 1 | 35 | 45 | – |
| 4 | 18 | 26 | – | 26 | 29 | 2 | 29 | 32 | 1 | 33 | 35 | – | 45 | 51 | – |

Total Elapsed Time = 51 minutes

Idle Time:

For machine $A = (51 - 26) = 25$ minutes

For machine $B = (51 - 29 + 7 + 2 + 2) = 33$ minutes

For machine $C = (51 - 32 + 12 + 2 + 3 + 1) = 37$ minutes

For machine $D = (51 - 35 + 14 + 4 + 1) = 35$ minutes

For machine $E = (51 - 51 + 18 + 1) = 19$ minutes

Now, the arithmetic mean $\left(\frac{G+H}{2}\right)$ value for each job on machines $G$ and $H$ is of the given table:

| Jobs | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Mean | 18 | 23 | 21.5 | 15 |

The required job sequence for the given table is:

| | | | |
|---|---|---|---|
| 1 | 3 | 2 | 4 |

Which is similar to the previous result, so the result remains unchanged.

The Harmonic mean ($HM_{av} = \frac{2}{\frac{1}{G}+\frac{1}{H}}$) value of the given table for each job on machines $G$ and $H$ is:

| Jobs | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Mean | 17.94 | 22.83 | 21.40 | 14.93 |

The required job sequence for the given table is:

| | | | |
|---|---|---|---|
| 1 | 3 | 2 | 4 |

Which is similar to the previous result, so the result remains unchanged.

The **Quadratic mean** $QM_{av} = \sqrt{\frac{1}{2}(G^2 + H^2)}$ of the given table for each job on machines $G$ and $H$ is:

| Jobs | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Mean | 18.03 | 23.09 | 21.55 | 15.03 |

The required job sequence for the given table is:

| | | | |
|---|---|---|---|
| 1 | 3 | 2 | 4 |

Which is similar to the previous result, so the result remains unchanged.

So, we can see that when we applied to a more complex 4-job, 5-machine problem, all the tested methods including Johnson's method and the mean based heuristics yielded identical results.

## V. Conclusion

This study presents a comparative exploration of job sequencing techniques using both classical and heuristic approaches, with a special focus on mean-based methods. By analyzing sequencing performance across Johnson's algorithm, the Greedy approach, and four different mean-based heuristics (Arithmetic, Harmonic, Quadratic, and Contra harmonic), the research demonstrates that while total elapsed time often remains consistent across techniques, idle time varies significantly depending on the method used.

The mean based methods provided solutions with a total elapsed time that was very close to the near optimal solution for every case even for n job m machine problems. This indicates that even in more complex scenarios, these simple heuristics are highly effective. The proposed mean based methods are not only effective but are robust and reliable. Providing results are consistent with established algorithms, even as the problem size and complexity increase. The Mean technique uniquely minimizes machine idle time, a key practical advantage.This approach is simple, computationally efficient, and reliable. It offers a valuable new tool for industrial practitioners and researchers.

Future work may extend the proposed Mean Processing Time Method to stochastic and dynamic job sequencing environments. It can be tested on large-scale, real-world datasets to assess scalability. Integration with meta heuristic algorithms such as Genetic Algorithms or Ant Colony Optimization could further enhance solution quality.

## Reference


1. Frederick S Hillier and J. Gerald Lieberman, Introduction to Operations Research.
2. Morse, P. M., & G. E. Kimball, 1951. Methods of operations research. MIT Press.
3. Blackett, P. M. S., 1962. *Studies of war: Operational research, 1939–45*. Oliver & Boyd.
4. McCloskey, J. F., & L. N.Trefethen, 1954. *Operations research for management*. Johns Hopkins Press.
5. Taha, H. A., 2017. *Operations Research: An Introduction* (10th ed.). Pearson.
6. Winston, W. L., 2004. *Operations Research: Applications and Algorithms* (4th ed.). Thomson Brooks/Cole.
7. Pinedo, M., 2022. Scheduling: Theory, Algorithms, and Systems (6th ed.). Springer.
8. French, Simon, 1982. *Sequencing and Scheduling: An Introduction to the Mathematics of the Job-Shop*. First ed., Ellis Horwood.
9. Pervaiz Iqbal, Dr.P.S. Sheik Uduman and Dr.S. Srinivasan, 2021. *Job Sequencing Problem Using Advanced Heuristics Techniques*. International Journal of Advanced Research in Engineering and Technology (IJARET).
10. Muhammad Nawaz, Emory Enscore Jr., and Inyong Ham, 1983. A Heuristic Algorithm for the m-Machine, n-Job Flow-shop Sequencing Problem.OMEGA – The International Journal of Management Science
11. Christos Koulamas, J.George Kyparises, 2007. "Single machine and two machine flow shop scheduling with general functions, European Journal of Operations Research.**178**,.402-407.
12. Kalczynski.P.J., J.Kamburowski, 2006. "A heuristic for minimizing the expected make span in two machine for flow shops with constant coefficient of variation", European Journal of Operations Research, **169,** .742-750.
13. Zhao. Y.F., L.X.Tang, 2007.Scheduling a single job continuous batch processing machine to minimize total completion time", Tien Tzu Hsuch pao/Acta Electronica Sinica,. **36,**. 367-370.
14. Pan.Y., L.Shi, H.Zhang, 2006. "A new solution approach to the general min max sequencing problem", Proceedings of the World Class on Intelligent Control and Automation(WCICA),.**1**., 1359-1364.

15. George Steiner, Paul Stephenson, 2007. Pare to optima for total weighted completion time and maximum lateness on a single machine", Discrete Applied Mathematics.

16. Drobouchevitch. I.G., V.A.Strusevich, 1999, A heuristics algorithm for two machines re-entrant shop scheduling", Annals of operation research.

17. Cai. X., V.Y.S.Lum, J.M.T.Cha, 1997. "Scheduling about a common due date with job dependent asymmetric earliness and tardiness penalties", European Journal of Operations Research.