DOI:10.26524/krj.2021.3

Kong. Res. J. 8 (1): 8-21, 2021 ISSN 2349-2694, All Rights Reserved,
Publisher: Kongunadu Arts and Science College, Coimbatore. https://www.krjournal.com

RESEARCH ARTICLE

INTERVAL SEQUENCING PROBLEM

Radhakrishnan, S.* and Saikeerthana, D.
Department of Mathematics, D.G. Vaishnav College (Autonomous), Chennai-600106, Tamil Nadu, India.

ABSTRACT

This paper deals with sequencing problems for 'n' jobs on single machine, 'n' jobs on two machines, 'n' jobs on three machines and 'n' jobs on 'm' machines. Here, we consider the sequencing problem where the processing time, due dates, weights are taken as intervals. An algorithm is provided for obtaining an optimal sequence and also for determining the minimum duration taken to complete all the jobs. To illustrate this, numerical examples are provided.

Keywords: Sequencing problem, Intervals, Optimal sequence, Total time elapsed, Idle time.

1. INTRODUCTION

Operations Research is relatively a new discipline which originated during World War II, and later became very popular throughout the world. It is used successfully in almost all the fields. Operations Research helps us to make better decisions in complex scenarios. It also includes the application of scientific tools for finding the optimum solution to a problem involving the operations of a system.

Sequencing problem is considered to be one of the important applications of Operations research. A series, in which a few jobs or tasks are to be performed following an order, is called sequencing. An algorithm was proposed by Johnson [1] for scheduling jobs in two machines. Its primary objective is to find an optimal sequence of jobs and to reduce the total amount of time it takes to complete all the jobs. It also reduces the amount of idle time between the two machines. Furthermore, Johnson's method has been extended to 'm' machines problem with an objective to complete all the jobs in a minimum duration.

Generally, in sequencing problems, the processing times are valued precisely. But in reality, it is perceived that the processing times during the performance of the job are imprecise and uncertain. In order to handle this uncertainties, we use fuzzy interval and fuzzy numbers. Here, we consider intervals. Interval computation was first suggested by Dwyer [2]. The concept of fuzzy sets was proposed by Zadeh [3]. Radhakrishnan et. al. [4] have solved problems on Game thoery using interval paramers. Radhakrishnan et. al. [5] have discussed and solved problems related to Critical Path Method and Programme Evaluation Review Technique with intervals and also with the conversion of fuzzy parameters (triangular and trapezoidal) into intervals using α-cuts.

The rest of this paper is organized as follows:
In section 2, basic preliminaries of interval and its arithmetic, types of intervals, ordering of intervals are given. In section 3, basic terminologies of sequencing and an algorithm for solving sequencing problem is provided. In section 4, numerical examples illustrating the algorithm are given. Finally, the conclusion.

2. PRELIMINARIES

2.1. Interval Number

An interval number A is defined as A = $[\beta_1, \beta_2]$ = $\{x: \beta_1 \leq x \leq \beta_2, x \in \mathbb{R}\}$. Here, $\beta_1, \beta_2 \in \mathbb{R}$ are the lower and upper bounds of the interval.

2.1.1. Arithmetic operations of interval

Let A = [β_1,β_2] and B= [γ_1,γ_2] be two intervals. Then

Addition: A+B =$[\beta_1 + \gamma_1, \beta_2 + \gamma_2]$

*Correspondence: Radhakrishnan, S., Department of Mathematics, D.G. Vaishnav College (Autonomous), Chennai-600106, Tamil Nadu, India. E.mail: dgvcradhakrishnan@gmail.com

Subtraction: A-B = $[\beta_1 - \gamma_2, \beta_2 - \gamma_1]$

Multiplication: A*B = $[\min(\beta_1\gamma_1, \beta_1\gamma_2, \beta_2\gamma_1, \beta_2\gamma_2), \max(\beta_1\gamma_1, \beta_1\gamma_2, \beta_2\gamma_1, \beta_2\gamma_2)]$

Division: $\frac{A}{B} = \frac{[\beta_1,\beta_2]}{[\gamma_1,\gamma_2]} = [\beta_1, \beta_2]. \frac{1}{[\gamma_1,\gamma_2]}$

where $\frac{1}{[\gamma_1,\gamma_2]} = \left[\frac{1}{\gamma_2}, \frac{1}{\gamma_1}\right], 0 \notin [\gamma_1, \gamma_2]$

$\frac{1}{[\gamma_1,0]} = \left[-\infty, \frac{1}{\gamma_1}\right]$

$\frac{1}{[0,\gamma_2]} = \left[\frac{1}{\gamma_2}, \infty\right]$ and

$\frac{1}{[\gamma_1,\gamma_2]} = \left[-\infty, \frac{1}{\gamma_1}\right] \cup \left[\frac{1}{\gamma_2}, \infty\right]$ = $[-\infty, \infty]$, $0 \in [\gamma_1, \gamma_2]$

Scalar Multiplication:

Let A= $[\beta_1, \beta_2]$ then uA= $[u\beta_1, u\beta_2]$, u≥ 0 and uA= $[u\beta_2, u\beta_1]$, u ≤ 0.

2.2. Types of intervals

Let A = $[\beta_1, \beta_2]$ and B= $[\gamma_1, \gamma_2]$ be two intervals. Therefore these can be classified into three types as follows:

Type I- Non overlapping intervals:

If two intervals are disjoint then they are known as non-overlapping intervals.

Type II- Partially overlapping intervals:

If one interval contains the other interval partially then they are known as partially overlapping intervals.

Type III- Completely overlapping intervals:

If one interval completely contained in the other interval then they are known as completely overlapping intervals.

These three types of intervals are shown in Figure 1

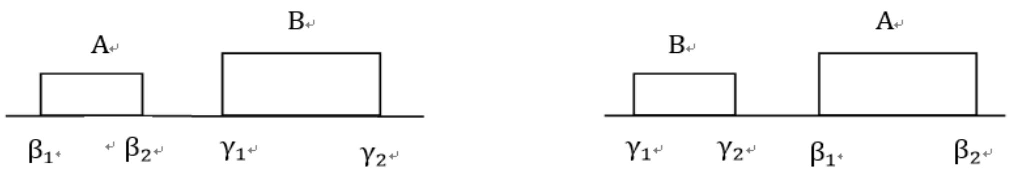

Figure 1(a): Type – I intervals

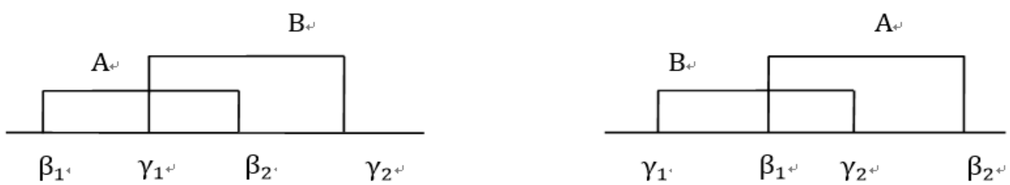

Figure 1(b): Type – II intervals

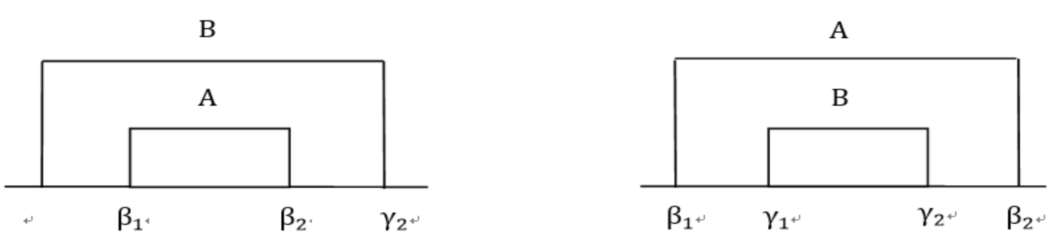

Figure 1(c): Type – III intervals

Figure 1: Different types of intervals

2.3. Ordering of intervals:

Let A= $[\beta_1, \beta_2]$ be the interval number. It can also be expressed by its centre and radius and it is denoted by $\langle a_C, a_W \rangle$, where $a_C = \frac{\beta_1+\beta_2}{2}$ and $a_W = \frac{\beta_2-\beta_1}{2}$ and they are known as centre and radius of the interval respectively.

Let A= $[\beta_1, \beta_2]$ = $\langle a_C, a_W \rangle$ and B =$[\gamma_1, \gamma_2]$ = $\langle b_C, b_W \rangle$. Then the relation on interval number is defined as

i. A < B iff $a_C < b_C$ whenever $a_C \neq b_C$.
ii. A > B iff $a_C > b_C$ whenever $a_C \neq b_C$.
iii. A < B iff $a_w < b_w$ whenever $a_C = b_C$.
iv. A > B iff $a_w > b_w$ whenever $a_C = b_C$.

3. Sequencing Problem

3.1. Principal assumptions:

While solving a sequencing problem, the following assumptions are made:

i. The processing times on different machines are not dependent of the order of the job in which they are to be performed.
ii. No machine can process more than one job concurrently.
iii. The time required in transferring a job from one machine to another machine is negligible and it is taken as zero.
iv. Each operation as well as job once started must be completed.
v. Processing times are known and fixed.
vi. An operation must be completed before its succeeding operation starts.

3.2. Basic Terminologies

a) Number of Machines:

Number of service facilities through which the job is to be passed.

b) Processing Time:

The time which is required for a job to process on a particular machine.

c) Processing Order:

It is a sequence in which various machines are needed for completion of the job.

d) Total elapsed time:

Total elapsed time is the total time required to complete the jobs from first to the last in a sequence.

e) Idle Time:

Idle time on a machine is the time for which the machine remains idle during the total elapsed time.

f) No passing rule:

If each of the n jobs is to be processed through two machines M_1 and M_2 in the order M_1 M_2 then it must go on machine M_1 first and then to Machine M_2.

3.3. Algorithm for Solving Interval sequencing Problem

3.3.1. Processing n jobs on single machine

Consider 'n' jobs processing on single machine where the processing time is given with the following objective:

1. Any new job which comes should not affect the processing time of these 'n' jobs.
2. If a new job comes, it has to wait for being considered in the next batch of jobs until the processing of the current 'n' jobs is completed.

This type of problem can be completely described as

i. Only single machine is involved.
ii. The processing time, due dates and weights are denoted as intervals.

Notations:

Let n_i - number of different jobs,
t_i - processing time of job i,
c_i - completion time of job i,
T_{c_i} - Total completion time of jobs,
d_i - due date of job i,
w_i - weight of job i,
L_i - Lateness of job i = c_i - d_i
S_i - Slack time of job i = d_i - t_i

Use the ordering of intervals to obtain an optimal sequence. The optimal sequence for an interval sequencing problem of a single machine can be determined by the following rules:

A. SPT Rule (Shortest Processing Time):

Jobs are arranged in ascending order of processing time.

B. WSPT Rule (Weighted Shortest Processing Time):

Jobs are arranged with minimum processing time per unit of importance in increasing order.

C. EDD Rule (Earliest Due Date):

Jobs are arranged in the increasing order of due dates of jobs.

D. Hodgson's Algorithm:

This Algorithm is applicable only if the number of late jobs is more than one. Number of late jobs can be identified using

EDD rule. Select the first late job and examine the longest processing time and remove it. Repeat the process until all the late jobs get vanished.

E. STR (Slack Time Remaining) Rule:

Jobs are arranged in the increasing order of Slack time.

After obtaining an optimal sequence for a single machine, we calculate the following:

i. Mean flow time = $\frac{\sum T_{c_i}}{\text{Number of jobs}}$

ii. Average in process inventory = $\frac{\sum(\text{Jobs waiting as in process inventory} * \text{range})}{\sum \text{Range}}$

where range = Time out of job n_i – Time in of job n_i

iii. Weighted Mean Flow time = $\frac{\sum_{i=1}^{n} w_i T_{c_i}}{\sum_{i=1}^{n} w_i}$

iv. Mean lateness = $\frac{\sum L_i}{\text{Number of jobs}}$

v. Number of late jobs = No. of positive lateness.

3.3.2. Processing 'n' jobs on two machines

Let A_1', A_2'... A_n' be the processing times of 'n' jobs on Machine 1 and B_1', B_2'...B_n' be the processing times of 'n' jobs on Machine 2. The problem is to find the order in which the 'n' jobs are to be processed through two machines with the minimum total elapsed time.

Procedure:

Step 1: Use ordering of intervals to identify the minimum processing time from the given list of processing times A_1', A_2' ...A_n' and B_1', B_2'... B_n'.

Step 2: If the minimum processing time is A_p' (i.e., job number p on machine 1) then do the p^{th} job first in the sequence. If the minimum processing time is B_q' (i.e., job number q on machine 2) then do the q^{th} job last in the sequence.

Step 3:

a) If there is a tie in minimum processing of both machines (i.e., $A_p' = B_q'$), process the p^{th} job first and q^{th} job last in the sequence.

b) If the tie for the minimum occurs among the processing time on Machine 1, select the job corresponding to the minimum of processing time on Machine 2 and process it first.

c) If the tie for the minimum occurs among the processing time on Machine 2, select the job corresponding to the minimum of processing time on Machine 1 and process it last.

Step 4: Cancel the jobs already assigned and repeat steps 2 to 4 until all the jobs have been assigned.

The resulting order will minimise the total elapsed time and it is known as optimal sequence.

Step 5: After obtaining an optimal sequence as stated above, the total elapsed time and also the idle time on machines 1 and 2 are calculated as follows:

Total elapsed time = Time out of the last job on machine 2.

Idle time for machine 1 = Total elapsed time - time when the last job is out of machine 1

Idle time for machine 2 = Time at which the first job on machine 1 finishes in a sequence $+\sum_{i=2}^{n}\left\{\begin{matrix}(\text{time when the } i^{th} \text{ job starts on machine 2}) \\ -(\text{time when the } (i-1)^{th} \text{ finishes on machine 2})\end{matrix}\right\}$

3.3.3. Processing 'n' jobs on three machines

Let A_1', A_2'... A_n' be the processing times of 'n' jobs on Machine 1, B_1', B_2'... B_n' be the processing times of 'n' jobs on Machine 2 and C_1', C_2'... C_n' be the processing times of 'n' jobs on Machine 3. There is no standard procedure to obtain an optimal sequence for processing 'n' jobs on 3 Machines. So, we have to convert the three machine problem into a two machine problem by satisfying any one or both of the following conditions.

1. Min (A_i') $\geqq$ Max (B_i'), for i= 1, 2 .... n
2. Min (C_i') $\geqq$ Max (B_i'), for i= 1, 2 .... n

To determine the minimum or maximum of processing time on machines, we use ordering of intervals.

If one of the above conditions is satisfied, we introduce two fictitious machines G and H such that the processing times on G and H are given by

$G = A_i' + B_i'$, for i= 1, 2... n

$H = B_i' + C_i'$, for i= 1, 2... n

Now we can proceed to determine the optimal sequence using 3.3.2.

After obtaining an optimal sequence, the total elapsed time and also the idle time on machines 1, 2 and 3 are calculated as follows:

Total elapsed time = Time out of the last job on machine 3.

Idle time for machine 1 = Total elapsed time - time when the last job is out of machine 1

Idle time for machine 2 = (Total elapsed time - time when the last job is out of machine 2)
+ Time at which the first job in a sequence finishes on machine 1

$$+\sum_{i=2}^{n}\begin{Bmatrix}(\text{time when the } i^{th} \text{ job starts on machine } 2)\\ -(\text{time when the } (i-1)^{th} \text{ finishes on machine } 2)\end{Bmatrix}$$

Idle time for machine 3 = Time at which the first job in a sequence finishes on machine 2

$$+\sum_{i=2}^{n}\begin{Bmatrix}(\text{time when the } i^{th} \text{ job starts on machine } 3)\\ -(\text{time when the } (i-1)^{th} \text{ finishes on machine } 3)\end{Bmatrix}$$

3.3.4. Processing 'n' jobs on 'm' machines

Let there be 'n' jobs which are to be processed through 'm' machines M_1, M_2 ...M_m in the order M_1, M_2 ...M_m and T_{ik} be the time taken by the i^{th} job on k^{th} machine.

Procedure

Step 1: Use ordering of intervals to identify Min T_{i1} (Minimum time for the first machine), Min T_{im} (Minimum time on the last machine) and Max (T_{ik}) for k=2, 3... m-1 and i=1, 2 ...n (Maximum time on intermediate machines).

Step 2: Check the following conditions:

(i) Minimum Time T_{i1} for the first machine (M_1) $\geq$ Maximum time (T_{ik}) on intermediate machines (M_2 to M_{m-1})

(ii) Minimum time T_{im} for the last machine (M_m) $\geq$ Maximum Time (T_{ik}) on intermediate machines (M_2 to M_{m-1}). (i.e., the minimum processing time on the machines M_1 and M_m (First and last machines) should be greater than or equal to maximum time on any of the 2 to m-1 machines).

Step 3: If the conditions in step 3 are not satisfied, the problem cannot be solved by this method, hence go to next step.

Step 4: Convert the 'n' job 'm' machine problem into 'n' job two machine problem by considering two machines G and H such that
$G_{ij} = T_{i1}+T_{i2}+ \ldots + T_{i(m-1)}$
$H_{ij} = T_{i2}+T_{i3}+ \ldots +T_{im}$

Step 5: Now we can proceed to determine the optimal sequence using 3.3.2. After obtaining an optimal sequence, the total elapsed time and also the idle time on machines are determined.

4. NUMERICAL EXAMPLE

1. Eight jobs A, B, C, D, E, F, G and H are to be processed on a single machine. The processing time, due dates, importance weights of the jobs are represented below. Assuming that no new jobs arrived thereafter, determine using SPT Rule, WSPT Rule, EDD Rule, STR Rule and Hodgson's Algorithm.

i. Optimal Sequence
ii. Completion time of the jobs
iii. Mean flow time as well as weighted mean flow time
iv. Average in process inventory
v. Lateness, mean lateness and maximum lateness
vi. Number of jobs actually late.

Jobs	A	B	C	D	E	F	G	H
Processing Time (t_i)	[3,7]	[6,10]	[5,7]	[1,5]	[8,12]	[13,15]	[4,10]	[1,5]
Due date (d_i)	[14,16]	[8,12]	[13,17]	[24,26]	[18,22]	[38,42]	[44,46]	[49,51]
Importance weight (w_i)	[0.5,1.5]	[1,3]	[1,5]	[0.5,1.5]	[1,3]	[1,5]	[1,3]	[0.5,1.5]

I. SPT Rule:

i. Use the ordering of intervals to obtain an optimal sequence. The optimal sequence is obtained by arranging the processing time in increasing order.
Optimal Sequence: D-H-A-C-G-B-E-F

ii. Completion time of jobs

Jobs	**D**	**H**	**A**	**C**	**G**	**B**	**E**	**F**
Time in	[0,0]	[1,5]	[2,10]	[5,17]	[10,24]	[14,34]	[20,44]	[28,56]
Time out	[1,5]	[2,10]	[5,17]	[10,24]	[14,34]	[20,44]	[28,56]	[41,71]

iii. Mean flow Time = $\frac{[1,5]+[2,10]+[5,17]+[10,24]+[14,34]+[20,44]+[28,56]+[41,71]}{8}$

$= \frac{[121,261]}{8}$ = [15.125, 32.625]

iv. Number of jobs waiting in process inventory are 8 during [0,0] - [1,5], 7 during [1,5] - [2,10], 6 during [2,10] - [5,17], 5 during [5,17] - [10,24], 4 during [10,24] - [14,34], 3 during [14,34] - [20,44], 2 during [20,44] - [28,56], 1 during [28,56] - [41,71].

Average in process inventory

$$= \frac{(8*[1,5])+(7*[-3,9])+(6*[-5,15])+(5*[-7,19])+(4*[-10,24])+(3*[-14,30])+(2*[-16,36])+(1*[-15,43])}{[1,5]+[-3,9]+[-5,15]+[-7,19]+[-10,24]+[-14,30]+[-16,36]+[-15,43]}$$

$$= \frac{[-207,589]}{[-69,181]}$$

= [-8.53, 3.25]

v. Lateness of various jobs are given by

Lateness of job D: [1, 5] - [24, 26] = [-25, -19]
Lateness of job H: [2, 10] - [49, 51] = [-49, -39]
Lateness of job A: [5, 17] - [14, 16] = [-11, 3]
Lateness of job C: [10, 24] - [13, 17] = [-7, 11]
Lateness of job G: [14, 34] - [44, 46] = [-32, -10]
Lateness of job B: [20, 44] - [8, 12] = [8, 36]
Lateness of job E: [28, 56] - [18, 22] = [6, 38]
Lateness of job F: [41, 71] - [38, 42] = [-1, 33]

Mean lateness = $\frac{[-111,53]}{8}$

= [-13.875, 6.625]

Maximum lateness = [8, 36] (job B) and [6, 38] (job E)

vi. Number of jobs actually late = 4.

II. WSPT Rule:

Jobs	Processing time (t_i)	Due date (d_i)	Importance weight (w_i)	$\frac{t_i}{w_i}$
A	[3,7]	[14,16]	[0.5,1.5]	[2,14]
B	[6,10]	[8,12]	[1,3]	[2,10]
C	[5,7]	[13,17]	[1,5]	[1,7]
D	[1,5]	[24,26]	[0.5,1.5]	$\left[\frac{1}{1.5}, 10\right]$
E	[8,12]	[18,22]	[1,3]	$\left[\frac{8}{3}, 12\right]$
F	[13,15]	[38,42]	[1,5]	$\left[\frac{13}{5}, 15\right]$
G	[4,10]	[44,46]	[1,3]	$\left[\frac{4}{3}, 10\right]$
H	[1,5]	[49,51]	[0.5,1.5]	$\left[\frac{1}{1.5}, 10\right]$

i. The Jobs are sequenced in increasing order of $\frac{t_i}{w_i}$.

Optimal sequence: C-D-H-G-B-E-A-F

ii. Completion time of the jobs

Job	C	D	H	G	B	E	A	F
Time in	[0,0]	[5,7]	[6, 12]	[7,17]	[11,27]	[17,37]	[25,49]	[28,56]
Time out	[5,7]	[6, 12]	[7,17]	[11,27]	[17,37]	[25,49]	[28,56]	[41,71]

iii. Mean flow time $= \frac{[140,276]}{8}$

$= [17.5, 34.5]$

Weighted Mean flow time

$$= \frac{([1,5]*[5,7])+([0.5,15]*[6,12])+([0.5,1.5]*[7,17])+([1,3]*[11,27])+([1,3]*[17,37]) +([1,3]*[25,49])+([0.5,1.5]*[28,56])+([1,5]*[41,71])}{[1,5]+[0.5,15]+[0.5,1.5]+[1,3]+[1,3]+[1,3]+[0.5,1.5]+[1,5]}$$

$$= \frac{[119.5,856.5]}{[6.5,23.5]}$$

$= [5.085, 131.76]$

iv. Number of jobs waiting in process inventory are 8 during [0,0] - [5,7], 7 during [5,7] - [6,12], 6 during [6,12] - [7,17], 5 during [7,17] - [11,27], 4 during [11,27] - [17,37], 3 during [17,37] - [25,49], 2 during [25,49] - [28,56], 1 during [28,56] - [41,71]

Average in inventory process

$$= \frac{(8*[5,7])+(7*[-1,7])+(6*[-5,11])+(5*[-6,20])+(4*[-10,26])+(3*[-12,32])+(2*[-21,31])+(1*[-15,43])}{[5,7]+[-1,7]+[5,11]+[-6,20]+[-10,26]+[-12,32]+[-21,31]+[-15,43]}$$

$$= \frac{[-160,576]}{[-65,177]}$$

= [-8.86, 3.25]

v. Lateness of various jobs are given by
Lateness of job C: [5, 7] - [13, 17] = [-12, -6]
Lateness of job D: [6, 12] - [24, 26] = [-20, -12]
Lateness of job H: [7, 17] - [49, 51] = [-44, -32]
Lateness of job G: [11, 27] - [44, 46] = [-35, -17]
Lateness of job B: [17, 37] - [8, 12] = [5, 29]
Lateness of job E: [25, 49] - [18, 22] = [3, 31]
Lateness of job A: [28, 56] - [14, 16] = [12, 42]
Lateness of job F: [41, 71] - [38, 42] = [-1, 33]

Mean lateness = $\frac{[-92,68]}{8}$ = [-11.5, 8.5]

Maximum lateness = [12, 42] (job A)

vi. Number of jobs actually late = 4

III. Slack Time Remaining Rule

Job	A	B	C	D	E	F	G	H
Processing time (t_i)	[3,7]	[6,10]	[5,7]	[1,5]	[8,12]	[13,15]	[4,10]	[1,5]
Due date (d_i)	[14,16]	[8,12]	[13,17]	[24,26]	[18,22]	[38,42]	[44,46]	[49,51]
Slack time (d_i-t_i)	[7,13]	[-2,6]	[6,12]	[19,25]	[6,14]	[23,29]	[34,42]	[44,50]

i. Use the ordering of intervals to obtain an optimal sequence. The optimal sequence is obtained by arranging the slack time in increasing order.
Optimal Sequence: B-C-A-E-D-F-G-H

ii. Completion time of these jobs

Job	B	C	A	E	D	F	G	H
Time in	[0,0]	[6,10]	[11,17]	[14,24]	[22,36]	[23,41]	[36,56]	[40,66]
Time out	[6,10]	[11,17]	[14,24]	[22,36]	[23,41]	[36,56]	[40,66]	[41,71]

iii. Mean Flow time = $\frac{[193,321]}{8}$ = [24.125, 40.125]

iv. Number of jobs waiting as in process inventory are 8 during [0,0]-[6,10], 7 during [6,10]-[11,17], 6 during [11,17]-[14,24], 5 during [14,24]-[22,36], 4 during [22,36]-[23,41], 3 during [23,41]-[36,56], 2 during [36,56]-[40,66], 1 during [40,66]-[41,71]

Average in process inventory

$$= \frac{(8*[6,10])+(7*[1,11])+(6*[-3,13])+(5*[-2,22])+(4*[-13,19])+(3*[-5,33])+(2*[-16,30])+(1*[-25,31])}{[6,10]+[1,11]+[-3,13]+[-2,22]+[-13,19]+[-5,33]+[-16,30]+[-25,31]}$$

$$= \frac{[-97,611]}{[-57,169]}$$

= [-10.72, 3.61]

v. Lateness of various jobs are given by

Lateness of job B: [6, 10] - [8, 12] = [-6, 2]
Lateness of job C: [11, 17] - [13, 17] = [-6, 4]
Lateness of job A: [14, 24] - [14, 16] = [-2, 10]
Lateness of job E: [22, 36] - [18, 22] = [0, 18]
Lateness of job D: [23, 41] - [24, 26] = [-3, 17]
Lateness of job F: [36, 56] - [38, 42] = [-6, 18]
Lateness of job G: [40, 66] - [44, 46] = [-6, 22]
Lateness of job H: [41, 71] - [49, 51] = [-10, 22]

Mean lateness = $\frac{[-39,113]}{8}$ = [-4.875, 14. 125]

Maximum lateness= [0, 18] (job E)

vi. Number of jobs actually late = 6.

IV. Earliest Due Date (EDD)

i. Use the ordering of intervals to obtain an optimal sequence. The optimal sequence is obtained by arranging the due dates of jobs in increasing order.
Optimal sequence: B-A-C-E-D-F-G-H

ii. Completion time of jobs

Job	B	A	C	E	D	F	G	H
In	[0,0]	[6,10]	[9,17]	[14,24]	[22,36]	[23,41]	[36,56]	[40,66]
Out	[6,10]	[9,17]	[14,24]	[22,36]	[23,41]	[36,56]	[40,66]	[41,71]

iii. Mean Flow time = $\frac{[191,321]}{8}$ = [23.875, 40.125]

iv. Number of jobs waiting as in process inventory are 8 during [0,0]-[6,10], 7 during [6,10]-[9,17], 6 during [9,17]-[14,24],5 during [14,24]-[22,36], 4 during [22,36]-[23,41],3 during [23,41]-[36,56],2 during [36,56]-[40,66], 1 during [40,66]-[41,71].

Average in process inventory

$$= \frac{(8*[6,10])+(7*[-1,11])+(6*[-3,15])+(5*[-2,22])+(4*[-13,19])+(3*[-5,33])+(2*[-16,30])+(1*[-25,31])}{[6,10]+[-1,11]+[-3,15]+[-2,22]+[-13,19]+[-5,33]+[-16,30]+[-25,31]}$$

$= \frac{[-103,623]}{[-59,171]}$

= [-10.56, 3.64]

v. Lateness of various jobs are given by
   Lateness of job B: [6, 10] - [8, 12] = [-6, 2]
   Lateness of job A: [9, 17] - [14, 16] = [-7, 3]
   Lateness of job C: [14, 24] - [13, 17] = [-3, 11]
   Lateness of job E: [22, 36] - [18, 22] = [0, 18]
   Lateness of job D: [23, 41] - [24, 26] = [-3, 17]
   Lateness of job F: [36, 56] - [38, 42] = [-6, 18]
   Lateness of job G: [49, 66] - [44, 46] = [-6, 22]
   Lateness of job H: [41, 71] - [49, 51] = [-10, 22]
   Mean lateness = $\frac{[-41,113]}{8}$ = [-5.125, 14.125]
   Maximum lateness = [0, 18] (job E)

vi. Number of jobs actually late = 6

V. Hodgson's Algorithm:

The Algorithm is applicable only if number of late jobs is more than 1.
Using EDD Rule, Number of late jobs could be found. As per EDD Rule, sequence of jobs will be B-A-C-E-D-F-G-H.

Job	Processing time (t_i)	Completion time(c_i)	Due date(d_i)	Lateness (c_i-d_i)
B	[6,10]	[6,10]	[8,12]	[-6,2]
A	[3,7]	[9,17]	[14,16]	[-7,3]
C	[5,7]	[14,24]	[13,17]	[-3,11]
E	[8,12]	[22,36]	[18,22]	[0,18]
D	[1,5]	[23,41]	[24,26]	[-3,17]
F	[13,15]	[36,56]	[38,42]	[-6,18]
G	[4,10]	[40,66]	[44,46]	[-6,22]
H	[1,5]	[41,71]	[49,51]	[-10,22]

Since job C is first late job and is in third position, examine first three jobs (B, A, C) to identify the one with longest processing time.
Here Job B has longest processing time of [6, 10]. Hence remove it and make the table again.

Job	Processing time (t_i)	Completion time(c_i)	Due date(d_i)	Lateness (c_i-d_i)
A	[3,7]	[3,7]	[14,16]	[-13,-7]
C	[5,7]	[8,14]	[13,17]	[-9,1]
E	[8,12]	[16,26]	[18,22]	[-6,8]
D	[1,5]	[17,31]	[24,26]	[-9,7]
F	[13,15]	[30,46]	[38,42]	[-12,8]
G	[4,10]	[34,56]	[44,46]	[-12,12]
H	[1,5]	[35,61]	[49,51]	[-16,12]

Now Job E is late. Since longest processing time up to Eth job (i.e.,) from Job A, C, E is job E. We will remove it.
Now, as per Hodgson's Algorithm
i. Optimal Sequence: A-C-D-F-G-H-B-E
ii. Completion time of these jobs.

Job	A	C	D	F	G	H	B	E
Time in	[0,0]	[3,7]	[8,14]	[9,19]	[22,34]	[26,44]	[27,49]	[33,59]
Time out	[3,7]	[8,14]	[9,19]	[22,34]	[26,44]	[27,49]	[33,59]	[41,71]

iii. Mean Flow time = $\frac{[169,297]}{8}$ = [21.125, 37.125]

iv. Number of jobs waiting as in process inventory are 8 during [0,0]-[3,7], 7 during [3,7]-[8,14], 6 during [8,14]-[9,19], 5 during [9,19]-[22,34],4 during [22,34]-[26,44], 3 during [26,44]-[27,49], 2 during [27,49]-[33,50], 1 during [33,59]-[41,71].

Average in process inventory

$= \frac{(8*[3,7])+(7*[1,11])+(6*[5,1])+(5*[3,25])+(4*[-8,22])+(3*[-17,23])+(2*[-16,32])+(1*[-18,38])}{[3,7]+[1,11]+[5,1]+[3,25]+[-8,22]+[-17,23]+[-16,32]+[-18,38]}$

$= \frac{[-57,523]}{[-47,159]}$

= [-11.13, 3.28]

v. Lateness of various jobs is given by

Lateness of job A: [3, 7] - [14, 16] = [-13,-7]
Lateness of job C: [8, 14] - [13, 17] = [-9, 1]
Lateness of job D: [9, 19] - [24, 26] = [-17, -5]
Lateness of job F: [22, 34] - [38, 42] = [-20, -4]
Lateness of job G: [26, 44] - [44, 46] = [-20, 0]
Lateness of job H: [27, 49] - [49, 51] = [-24, 0]
Lateness of job B: [33, 59] - [8, 12] = [21, 51]
Lateness of job E: [41, 71] - [18, 22] = [19, 53]

Mean lateness = $\frac{[-63,89]}{8}$= [-7.875, 11.125]

Maximum lateness = [21, 51] (job B) and [19, 53] (job E)

vi. Number of actually late jobs = 2.

2. Consider an interval sequencing problem for 5 jobs on 2 Machines with the processing time as intervals are given in the following table.

Jobs	A	B	C	D	E
Machine M_1	[1,5]	[7,9]	[3,7]	[6,8]	[2,6]
Machine M_2	[2,6]	[9,11]	[5,7]	[3,7]	[7,9]

Solution:

To obtain an optimal sequence:

Min(M_1)	Min(M_2)	Min (M_1 , M_2)	Optimal Sequence				
[1,5]	[2,6]	[1,5]	A	-	-	-	-
[2,6]	[3,7]	[2,6]	A	E	-	-	-
[3,7]	[3,7]	[3,7]	A	E	C	-	D
[7,9]	[9,11]	[7,9]	A	E	C	B	D

Optimal Sequence: A-E-C-B-D

To obtain the minimum total elapsed time:

Job optimal sequence	Machine M_1		Machine M_2		Idle time (M_2)
	Time in	Time out	Time in	Time out	
A	[0,0]	[1,5]	[1,5]	[3,11]	[1,5]
E	[1,5]	[3,11]	[3,11]	[10,20]	[-8,8]
C	[3,11]	[6,18]	[10,20]	[15,27]	[-10,10]
B	[6,18]	[13,27]	[15,27]	[24,38]	[-12,12]
D	[13,27]	[19,35]	[24,38]	[27,45]	[-14,14]
Total					[-43,49]

Minimum Total Elapsed Time = [27, 45]
Idle time for Machine M_1 = [27, 45] – [19, 35] = [-8, 26]
Idle time for Machine M_2 = [-43, 49]
3. Consider an interval sequencing problem for 7 jobs on 3 Machines with the processing time as intervals are given in the following table.

Jobs	A	B	C	D	E	F	G
Machine 1	[1,5]	[7,9]	[6,8]	[2,6]	[8,10]	[7,9]	[6,8]
Machine 2	[2,6]	[1,5]	[1,3]	[3,7]	[0,2]	[2,6]	[1,5]
Machine 3	[4,8]	[6,8]	[3,7]	[10,12]	[3,7]	[4,8]	[11,13]

Solution:
Since the problem is a 3 machine problem. We convert it into a 2 machine problem. For that it has to satisfy any one of the following conditions:

i. Min (M_1)$\geqq$ Max (M_2)
ii. Min (M_3)$\geqq$ Max (M_2)

Here Min (M_1) = [1, 5]; Max (M_2) = [3, 7] = Min (M_3)

i. Min (M_1) $\ngeqq$ Max (M_2)
ii. Min (M_3) = Max (M_2)

Therefore, the second condition is satisfied.
We convert this problem into 2 machine problem as H and K.
H = M_1 + M_2 and K = M_2 + M_3
The processing time of the 2 machines H and K for 7 jobs are as follows:

Machines	Jobs						
	A	B	C	D	E	F	G
H	[3,11]	[8,14]	[7,11]	[5,13]	[8,12]	[9,15]	[7,13]
K	[6,14]	[7,13]	[4,10]	[13,19]	[3,9]	[6,14]	[12,18]
Order of Cancellation	(2)	(6)	(3)	(4)	(1)	(7)	(5)

Optimal Sequence: A-D-G-F-B-C-E
To obtain Total Time Elapsed

Jobs	Machine M_1		Idle Time (M_1)	Machine M_2		Idle Time (M_2)	Machine M_3		Idle Time (M_3)
	Time in	Time out		Time in	Time out		Time in	Time out	
A	[0,0]	[1,5]	-	[1,5]	[3,11]	[1,5]	[3,11]	[7,19]	[3,11]
D	[1,5]	[3,11]	-	[3,11]	[6,18]	[-8,8]	[7,19]	[17,31]	[-12,12]
G	[3,11]	[9,19]	-	[9,19]	[10,24]	[-9.13]	[17,31]	[28,44]	[-14,14]
F	[9,19]	[16,28]	-	[16,28]	[18,34]	[-8,18]	[28,44]	[32,52]	[-16,16]
B	[16,28]	[23,37]	-	[23,37]	[24,32]	[-11,19]	[32,52]	[38,60]	[-20,20]
C	[23,37]	[29,45]	-	[29,45]	[30,48]	[-13,21]	[38,60]	[41,67]	[-22,22]
E	[29,45]	[37,55]	-	[37,55]	[37,57]	[-11,25]	[41,67]	[44,74]	[-26,26]
			[-11, 37]			[-13,37]			
Total			[-11, 37]			[-72,146]			[-107,121]

Minimum Total Elapsed Time = [44, 74]
Idle Time on Machine M_1 = [-11, 37]
Idle Time on Machine M_2 = [-72, 146]
Idle Time on Machine M_3 = [-107, 121]

4. Consider an interval sequencing problem for 4 jobs on 4 Machines with the processing time as intervals are given in the following table.

Jobs	Machines			
	M_1	M_2	M_3	M_4
A	[9,17]	[6,10]	[4,10]	[12,16]
B	[10,14]	[5,7]	[6,10]	[18,20]
C	[8,10]	[4,10]	[6,10]	[14,16]
D	[6,10]	[3,7]	[4,8]	[14,16]

Solution:
To find an Optimal Sequence, we convert the 4 Machine problem into a 2 Machine problem using Proposed Algorithm.

For this it has to satisfy any one of the following condition.

i) Min (M_1) ≥ Max (M_2 , M_3)
ii) Min (M_4) ≥ Max (M_2 , M_3)

Here Min (M_1) = [6, 10]; Min (M_4) = [12, 16];
Max (M_2) = [6, 10] = Max (M_3)
Max (M_2, M_3) = [6, 10]

Hence we have,

i) Min (M_1) = Max (M_2, M_3)
ii) Min (M_4) ≥ Max (M_2, M_3)

Here both the conditions are satisfied. We convert this problem into 2 machine problem as G and H.
G = M_1+ M_2+ M_3
H = M_2+ M_3+ M_4

Machines	Jobs			
	A	B	C	D
G	[19,37]	[21,31]	[18,30]	[13,25]
H	[22,36]	[29,37]	[24,36]	[21,31]
Order of Cancellation	(4)	(3)	(2)	(1)

Optimal Sequence: D-C-B-A
To Find Total time Elapsed:

Job	Machine M_1		Idle Time (M_1)	Machine M_2		Idle Time (M_2)	Machine M_3		Idle Time (M_3)	Machine M_4		Idle Time (M_4)
	Time in	Time out		Time In	Time out		Time in	Time out		Time in	Time out	
D	[0,0]	[6,10]	-	[6,10]	[9,17]	[6,10]	[9,17]	[13,25]	[9,17]	[13,25]	[27,41]	[13,25]
C	[6,10]	[14,20]	-	[14,20]	[18,30]	[-3,11]	[18,30]	[24,40]	[-7,17]	[27,41]	[41,57]	[-14,14]
B	[14,20]	[24,34]	-	[24,34]	[29,41]	[-6,16]	[29,41]	[35,51]	[-11,17]	[41,57]	[59,77]	[-16,16]
A	[24,34]	[33,51]	-	[33,51]	[39,61]	[-8,22]	[39,61]	[43,71]	[-12,26]	[59,77]	[71,93]	[-18,18]
						[-11,59]			[-21,77]			
			[20,60]			[10,54]			[0,50]			
Total			[20,60]			[-1,113]			[-21,127]			[-35,73]

Total Time Elapsed = [71, 93]
Idle time of Machine M_1 = [20, 60]
Idle time of Machine M_2 = [-1, 113]
Idle time of Machine M_3 = [-21, 127]
Idle time of Machine M_4 = [-35, 73]

Conclusion

In this paper, we discussed sequencing problem for ‘n’ jobs on single machine, ‘n’ jobs on two machines, ‘n’ jobs on three machines and ‘n’ jobs on ‘m’ machines. The processing time, due date, weights are considered as imprecise numbers and are described as intervals which are more realistic and general in nature. The numerical illustrations are more efficient to obtain an optimal sequence, completion time of jobs and total time elapsed time to process all jobs through machines. It helps to formulate uncertainty in actual environment and also serves as application for the decision makers in real life situation.

References

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

[2] Dwyer, P.S. (1951). Linear Computations, John Wiley and Sons, New York.

[3] Zadeh, L.A. (1965). Fuzzy sets, Information and Control, John Wiley and Sons, New York .vol. 8: 338 – 353.

[4] Radhakrishnan, S., Saikeerthana, D. (2020). Game theory problems using Interval parameters, Kong. Res. J. 7(2): 123-129.

[5] Radhakrishnan, S., Saikeerthana, D. (2020). Interval Network Analysis in project Management. Kong. Res. J. 7(2): 99-113.

About The License

Attribution 4.0 International (CC BY 4.0)

The text of this article is licensed under a Creative Commons Attribution 4.0 International License