Journal of Energy Technologies and Policy
ISSN 2224-3232 (Paper) ISSN 2225-0573 (Online)
Vol.1, No.3, 2011

Solving n-Jobs, 3-Machines Constrained Flow Shop Scheduling Processing Time, Setup Time Each Associated with Probabilities Including Job-Block Criteria

Sameer Sharma[1]

Department of Mathematics, Maharishi Markandeshwar University, Mullana, Ambala, India

* E-mail of the corresponding author: samsharma31@yahoo.com

Abstract

This paper is pertain to heuristic technique for n-jobs, 3-machines flowshop scheduling problem in which processing times and setup times are associated with their respective probabilities involving transportation time, break down interval and job block criteria is taken in to account. Further jobs are attached with weights to indicate their relative importance. The proposed method is very easy to understand and also provide an important tool for decision makers. A numerical illustration followed by a computer programme is also given to clarify the algorithm.

Keywords: Flow shop scheduling, Processing time, Setup time, Transportation time, Break down, Weights of job, Job block.

1. Introduction

During the last 30 years, the flow shop sequencing problems has been the center of attention of many researchers. Since Johnson had proposed optimal two and three stage production schedules, many heuristics approaches have been suggested to solve the various problems. The flow shop scheduling problem is a production scheduling problem in which each of the n jobs (tasks) must be processed in the same sequence on each one of m machines (processors). The scheduling problem practically depends upon the important factors namely, Job transportation which includes loading time, moving time and unloading time etc., Weightage of job which represents the relative importance of one job over another and Breakdown of machine which is due to failure of electric current or due to non supply of raw material or any other technical interruptions. The majority of scheduling research assumes setup which includes work to prepare the machine as negligible or part of the processing time while this assumption adversely affects solution quality for many applications which require explicit treatment of setup. Johnson (1954) proposed the well known Johnson's rule in the two stage flow shop scheduling problem. Yoshida & Hitomi (1979) further considered the problem with setup times. The work was developed by Belman (1956), Maggu & Das (1977), Miyazaki & Nishiyama (1980), Nawaz *et al* (1983), Singh (1985), Chandramouli (2005), Belwal & Mittal (2008), Khodadadi (2008), Pandian & Rajendran (2010), Gupta & Sharma (2011) by considering various parameters.

Gupta, Sharma & seema (2011) studied a n x 3 flowshop scheduling problem, processing time associated with probabilities involving transportation time, breakdown interval, Weightage of jobs and job block criteria. This paper is an attempt to extend the study made by Gupta & Sharma (2011) by introducing the concept of independent setup time with their corresponding probabilities. We have obtained an algorithm which minimize the total elapsed time whenever men weighted production flow time is taken into consideration.

2. Practical Situation

Many applied and experimental situations exist in our day-to-day working in factories and industrial production concerns etc. The practical situation may be taken in a paper mill, sugar factory and oil refinery etc. where various qualities of paper, sugar and oil are produced with relative importance i.e. weight in jobs.

In many manufacturing companies different jobs are processed on various machines. These jobs are required to process in a machine shop A, B, C, ---- in a specified order. When the machines on which jobs are to be processed are planted at different places, the transportation time (which includes loading time, moving time and unloading time etc.) has a significant role in production concern. Setup includes work to prepare the machine, process or bench for product parts or the cycle. This includes obtaining tools, positioning work-in-process material, return tooling, cleaning up, setting the required jigs and fixtures, adjusting tools and inspecting material and hence significant. The break down of the machines (due to delay in material, changes in release and tails date, tool unavailability, failure of electric current, the shift pattern of the facility, fluctuation in processing times, some technical interruption etc.) have significant role in the production concern. The idea of job block has practical significance to create a balance between a cost of providing priority in service to the customer and cost of giving service with non priority, .i.e. how much is to be charged from the priority customer(s) as compared to non priority customer(s).

3. Notations

S : Sequence of jobs 1, 2, 3… n

S_k : Sequence obtained by applying Johnson's procedure, k = 1, 2, 3, -------

M_j : Machine j, j= 1, 2, 3

M : Minimum makespan

a_{ij} : Processing time of i^{th} job on machine M_j

p_{ij} : Probability associated to the processing time a_{ij}

s_{ij} : Set up time of i^{th} job on machine M_j

q_{ij} : Probability associated to the set up time s_{ij}

A_{ij} : Expected processing time of i^{th} job on machine M_j

S_{ij} : Expected set up time of i^{th} job on machine M_j

β : Equivalent job for job – block

$A_{ij}^{'}$: Expected processing time of i^{th} job after break-down effect on j^{th} machine

$I_{ij}(S_k)$: Idle time of machine M_j for job i in the sequence S_k

$T_{i,j\rightarrow k}$: Transportation time of i^{th} job from j^{th} machine to k^{th} machine

w_i : weight assigned to i^{th} job

L : Length of break down interval.

4. Problem Formulation

Let some job i (i = 1,2,……..,n) is to be processed on three machines M_j (j = 1,2,3). Let a_{ij} be the processing time of i^{th} job on j^{th} machine with probabilities p_{ij} and s_{ij} be the setup time of i^{th} job on j^{th} machine with probabilities q_{ij}. Let $T_{i,j\rightarrow k}$ be the transportation time of i^{th} job from j^{th} machine to k^{th} machine. Let w_i be the weights assigned to the i^{th} job. Our aim is to find a sequence $\{S_k\}$ of the jobs which minimize total elapsed time, and weighted mean-flow times whenever mean weighted production flow time is taken into consideration.

The mathematical model of the problem in matrix form can be stated as:

Jobs	Machine A				$T_{i,1\to 2}$	Machine B				$T_{i,2\to 3}$	Machine C				Weights of jobs
i	a_{i1}	p_{i1}	s_{i1}	q_{i1}		a_{i2}	p_{i2}	s_{i2}	q_{i2}		a_{i3}	p_{i3}	s_{i3}	q_{i3}	w_i
1	a_{11}	p_{11}	s_{11}	q_{11}	$T_{1,1\to 2}$	a_{12}	p_{12}	s_{12}	q_{12}	$T_{1,2\to 3}$	a_{13}	p_{13}	s_{13}	q_{13}	w_1
2	a_{21}	p_{21}	s_{21}	q_{21}	$T_{2,1\to 2}$	a_{22}	p_{22}	s_{22}	q_{22}	$T_{2,2\to 3}$	a_{23}	p_{23}	s_{23}	q_{23}	w_2
3	a_{31}	p_{31}	s_{31}	q_{31}	$T_{3,1\to 2}$	a_{32}	p_{32}	s_{32}	q_{32}	$T_{3,2\to 3}$	a_{33}	p_{33}	s_{33}	q_{33}	w_3
4	a_{41}	p_{41}	s_{41}	q_{41}	$T_{4,1\to 2}$	a_{42}	p_{42}	s_{42}	q_{42}	$T_{4,2\to 3}$	a_{43}	p_{43}	s_{43}	q_{43}	w_4
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
n	a_{n1}	p_{n1}	s_{n1}	q_{n1}	$T_{n,1\to 2}$	a_{n2}	p_{n2}	s_{n2}	q_{n2}	$T_{n,2\to 3}$	a_{n3}	p_{n3}	s_{n3}	q_{n3}	w_n

Table 1

5. Algorithm

Step 1: Calculate the expected processing times and expected set up times as follows

$A_{ij} = a_{ij} \times p_{ij}$ and $S_{ij} = s_{ij} \times q_{ij}$ $\quad \forall i, j = 1,2,3$

Step 2: Check the condition

Either Max $\{A_{i1} + T_{i,1\to 2} - S_{i2}\} \geq$ Min$\{A_{i2} + T_{i,1\to 2} - S_{i1}\}$

or Max$\{A_{i3} + T_{i,2\to 3} - S_{i2}\} \geq$ Min$\{A_{i2} + T_{i,2\to 3} - S_{i3}\}$ or both for all i

If the conditions are satisfied then go to step 3, else the data is not in the standard form.

Step 3: Introduce the two fictitious machines G and H with processing times G_i and H_i as defined below:

$G_i = \left| A_{i1} - A_{i2} - T_{i,1\to 2} - T_{i,2\to 3} - \max(S_{i1}, S_{i2}) \right|$ and $H_i = \left| A_{i3} - A_{i2} - T_{i,1\to 2} - T_{i,2\to 3} + S_{i3} \right|$.

Step 4: Compute Minimum (G_i ,H_i)

If Min (G_i , H_i)=G_i then define $G_i^{'}$ =G_i+ w_i and $H_i^{'}$ =H_i .

If Min (G_i , H_i)=H_i then define $G_i^{'}$ =G_i and $H_i^{'}$ =H_i+ w_i .

Step 5: Define a new reduced problem with $G_i^{"}$ and $H_i^{"}$ where

$G_i^{"} = G_i^{'}/w_i \, , H_i^{"} = H_i^{'}/w_i \quad \forall i = 1,2,3.....,n$

Step 6: Find the expected processing time of job block β = (k, m) on fictitious machines G & H using equivalent job block criterion given by Maggu & Das (1977). Find $G"_\beta$ and $H"_\beta$ using $G"_\beta = G"_k + G"_m - \min(G"_m, H"_k)$ and $H"_\beta = H"_k + H"_m - \min(G"_m, H"_k)$

Step7: Define new reduced problem with processing time $G_i^{"}$ & $H_i^{"}$ as defined in step 5 and replace job block β = (k, m) by a single equivalent job β with processing times $G"_\beta$ & $H"_\beta$ as defined in step 6

Step 8: Using Johnson's procedure, obtain all sequences S_k having minimum elapsed time. Let these be S_1, S_2,....,S_r

Step 9: Prepare In-Out tables for the sequences $S_1, S_2, ..., S_r$ obtained in step 8. Let the mean flow time is minimum for the sequence S_k. Now, read the effect of break down interval (a, b) on different jobs on the lines of *Singh T.P.* (1985) for the sequence S_k.

Step 10: Form a modified problem with processing time A'_{ij} ; $i = 1,2,3,\ldots,n;\ j= 1, 2,3$

If the break down interval (a, b) has effect on job *i* then

$A'_{ij} = A_{ij} + L$; Where L = b – a, the length of break-down interval

If the break-down interval (a, b) has no effect on ith job then $A'_{ij} = A_{ij}$.

Step 11: Repeat the procedure to get the optimal sequence for the modified scheduling problem using steps 3 to step 9. Determine the total elapsed time.

Step 12: Find the performance measure studied in weighted mean flow time defined by $F = \sum_{i=1}^{n} w_i f_i \Big/ \sum_{i=1}^{n} w_i$, where f_i is flow time of i^{th} job.

6. **Programme**

```
#include<iostream.h>
#include<stdio.h>
#include<conio.h>
#include<process.h>
#include<math.h>

int n,j;
float a1[16],b1[16],c1[16],a11[16],b11[16],c11[16],g[16],h[16],T12[16],T23[16],s11[16],s22[16],s33[16];
float macha[16],machb[16],machc[16],macha1[16],machb1[16],machc1[16],maxs[16];
int f=1;float minval,minv,maxv1[16],maxv2[16],minv1;float w[16];
int group[2];//variables to store two job blocks
int bd1,bd2;// Breakdown interval
float gbeta=0.0,hbeta=0.0;float gbeta1=0.0,hbeta1=0.0;
void main()
{
	clrscr();
	int a[16],b[16],c[16],j[16],s1[16],s2[16],s3[16];float p[16],q[16],r[16],x[16],t1[16],u[16];
	cout<<"How many Jobs (<=15) : ";cin>>n;
	if(n<1 || n>15)
	{cout<<endl<<"Wrong input, No. of jobs should be less than 15..\n Exitting";getch();exit(0);}
	for(int i=1;i<=n;i++){j[i]=i;
cout<<"\nEnter the processing time, set up time and the probabilities of "<<i<<" job for machine A and Transportation time from Machine A to B : ";cin>>a[i]>>p[i]>>s1[i]>>x[i]>>T12[i];
cout<<"\nEnter the processing time, setup time and the probabilities of "<<i<<" job for machine B and Transportation time from Machine B to C : ";cin>>b[i]>>q[i]>>s2[i]>>t1[i]>>T23[i];
cout<<"\nEnter the processing time and its probability of "<<i<<"job for machine C: ";cin>>c[i]>>r[i]>>s3[i]>>u[i];
cout<<"\nEnter the weightage of "<<i<<"job:";cin>>w[i];
//Calculate the expected processing & setup times of the jobs for the machines:
a1[i] = a[i]*p[i];b1[i] = b[i]*q[i];c1[i] = c[i]*r[i];s11[i]=s1[i]*x[i]; s22[i]= s2[i]*t1[i]; s33[i]= s3[i]*u[i];}
cout<<endl<<"Expected processing time of machine A, B and C with weightage: \n";
for(i=1;i<=n;i++)
{cout<<j[i]<<"\t"<<a1[i]<<"\t"<<s11[i]<<"\t"<<T12[i]<<"\t"<<b1[i]<<"\t"<<s22[i]<<"\t"<<T23[i]<<"\t"<<c1[i]<<"\t"<<s33[i]<<"\t"<<w[i];cout<<endl;}
cout<<"\nEnter the two breakdown interval:";cin>>bd1>>bd2;
//Findinglargest in a1
```

```
float maxa1;maxa1=a1[1]+T12[1]-s22[1];
for(i=2;i<n;i++)
{if((a1[i]+T12[i]-s22[i])>maxa1)maxa1=a1[i]+T12[i]-s22[i];}
//For finding smallest in b1
float minb1;minb1=b1[1]+T12[1]-s11[1];
for(i=2;i<n;i++)
{if(b1[i]+T12[i]-s11[i]<minb1)minb1=b1[i]+T12[i]-s11[i];}
float minb2;minb2=b1[1]+T23[1]-s33[i];
for(i=2;i<n;i++)
{if((b1[i]+T23[i]-s33[i])<minb2)minb2=b1[i]+T23[i]-s33[i];}
//Finding largest in c1
float maxc1;maxc1=c1[1]+T23[1]-s22[i];
for(i=2;i<n;i++)
{if((c1[i]+T23[i]-s22[i])>maxc1)maxc1=c1[i]+T23[i]-s22[i];}
for(i=1;i<=n;i++)
    {if(s11[i]>s22[i]){maxs[i]=s11[i];}
else {maxs[i]=s22[i];}}
if(maxa1>=minb1||maxc1>=minb2)
{g[i]=fabs((a1[i]-T12[i]-b1[i]-T23[i]-maxs[i]));h[i]=fabs((c1[i]-T12[i]-b1[i]-T23[i]+s33[i]));}
else
{cout<<"\n data is not in Standard Form...\nExitting";getch();exit(0);}
for(i=1;i<=n;i++)
{g[i]=fabs(a1[i]-T12[i]-b1[i]-T23[i]-maxs[i]);h[i]=fabs(c1[i]-T12[i]-b1[i]-T23[i]+s33[i]);}
cout<<endl<<"Expected processing time for two fictious machines G and H: \n";
for(i=1;i<=n;i++)
 {cout<<endl;cout<<j[i]<<"\t"<<g[i]<<"\t"<<h[i]<<"\t"<<w[i];cout<<endl;}
//To find minimum of G & H
float g1[16],h1[16];
for (i=1;i<=n;i++)if(g[i]<=h[i]){g1[i]=g[i]+w[i];h1[i]=h[i];}
else{g1[i]=g[i];h1[i]=h[i]+w[i];}
float g2[16],h2[16];
for(i=1;i<=n;i++)
{g2[i]=g1[i]/w[i];h2[i]=h1[i]/w[i];}
cout<<endl<<endl<<"displaying original scheduling table"<<endl;
for(i=1;i<=n;i++)
{cout<<j[i]<<"\t"<<g2[i]<<"\t"<<h2[i]<<endl;}
cout<<"\nEnter the two job blocks(two numbers from 1 to "<<n<<"):";    cin>>group[0]>>group[1];
 //calculate G_Beta and H_Beta
if(g2[group[1]]<h2[group[0]])
{minv=g2[group[1]];}
else{minv=h2[group[0]];}
gbeta=g2[group[0]]+g2[group[1]]-minv;hbeta=h2[group[0]]+h2[group[1]]-minv;
cout<<endl<<endl<<"G_Beta="<<gbeta;cout<<endl<<"H_Beta="<<hbeta;
int j1[16];float g13[16],h13[16];
for(i=1;i<=n;i++)
{if(j[i]==group[0]||j[i]==group[1]){f--;}
else{j1[f]=j[i];}f++;}
j1[n-1]=17;
for(i=1;i<=n-2;i++)
{g13[i]=g2[j1[i]];h13[i]=h2[j1[i]];}
g13[n-1]=gbeta;h13[n-1]=hbeta;
cout<<endl<<endl<<"displaying original scheduling table"<<endl;
```



```
for(i=1;i<=n-1;i++)
{cout<<j1[i]<<"\t"<<g13[i]<<"\t"<<h13[i]<<endl;}
float mingh[16];char ch[16];
for(i=1;i<=n-1;i++)
   {if(g13[i]<h13[i])
         {mingh[i]=g13[i];ch[i]='g';}
    else {mingh[i]=h13[i];ch[i]='h'; }}
        for(i=1;i<=n-1;i++)

        for(int j=1;j<=n-1;j++)
          {if(mingh[i]<mingh[j])
        {float temp=mingh[i]; int temp1=j1[i]; char d=ch[i];mingh[i]=mingh[j]; j1[i]=j1[j]; ch[i]=ch[j];
        mingh[j]=temp; j1[j]=temp1; ch[j]=d;}}
// calculate beta scheduling
float sbeta[16];int t=1,s=0;
for(i=1;i<=n-1;i++)
    {if(ch[i]=='h'){ sbeta[(n-s-1)]=j1[i];s++;}
else if(ch[i]=='g'){sbeta[t]=j1[i];t++;}}
int arr1[16], m=1;cout<<endl<<endl<<"Job Scheduling:"<<"\t";
for(i=1;i<=n-1;i++)
    {if(sbeta[i]==17){arr1[m]=group[0];arr1[m+1]=group[1];
    cout<<group[0]<<" "<<group[1]<<" ";m=m+2;continue;}
else     {cout<<sbeta[i]<<" ";arr1[m]=sbeta[i];m++;}}
//calculating total computation sequence
  float time=0.0,macha11[16];macha[1]=time+a1[arr1[1]];
    for(i=2;i<=n;i++)
    {macha11[i]=macha[i-1]+s11[arr1[i-1]];macha[i]=macha11[i]+a1[arr1[i]];}
    machb[1]=macha[1]+b1[arr1[1]]+T12[arr1[1]];
for(i=2;i<=n;i++)
    {if((machb[i-1]+s22[arr1[i-1]])>(macha[i]+T12[arr1[i]]))maxv1[i]=machb[i-1]+s22[arr1[i-1]];
else maxv1[i]=macha[i]+T12[arr1[i]];machb[i]=maxv1[i]+b1[arr1[i]];}
    machc[1]=machb[1]+c1[arr1[1]]+T23[arr1[1]];
for(i=2;i<=n;i++)
    {if((machc[i-1]+s33[arr1[i-1]])>(machb[i]+T23[arr1[i]]))
maxv2[i]=machc[i-1]+s33[arr1[i-1]];
else
    maxv2[i]=machb[i]+T23[arr1[i]];machc[i]=maxv2[i]+c1[arr1[i]];}
cout<<endl<<endl<<"In-Out Table is:"<<endl<<endl;
cout<<"Jobs"<<"\t"<<"Machine M1"<<"\t"<<"\t"<<"Machine M2" <<"\t"<<"\t"<<"Machine M3"<<endl;
cout<<arr1[1]<<"\t"<<time<<"--"<<macha[1]<<"
\t"<<"\t"<<macha[1]+T12[arr1[1]]<<"--"<<machb[1]<<"
\t"<<"\t"<<machb[1]+T23[arr1[1]]<<"--"<<machc[1]<<endl;
if((time<=bd1 && macha[1]<=bd1)||(time>=bd2 && macha[1]>=bd2))
    {a1[arr1[1]]=a1[arr1[1]];}
else
    {a1[arr1[1]]+=(bd2-bd1);}
if((macha[1]+T12[arr1[1]])<=bd1 && machb[1]<=bd1||(macha[1]+T12[arr1[1]])>=bd2 &&
machb[1]>=bd2)
    {b1[arr1[1]]=b1[arr1[1]];}
else
    {b1[arr1[1]]+=(bd2-bd}
if((machb[1]+T23[arr1[1]])<=bd1 && machc[1]<=bd1||(machb[1]+T23[arr1[1]])>=bd2 &&
```

```
machc[1]>=bd2)
	{c1[arr1[1]]=c1[arr1[1]];}
else
	{c1[arr1[1]]+=(bd2-bd1);}for(i=2;i<=n;i++)
	{cout<<arr1[i]<<"\t"<<macha11[i]<<"--"<<macha[i]<<"	"<<"\t"<<maxv1[i]<<"--"<<machb[i]<<"
"<<"\t"<<maxv2[i]<<"--"<<machc[i]<<endl;
if(macha11[i]<=bd1 && macha[i]<=bd1 || macha11[i]>=bd2 && macha[i]>=bd2)
	{a1[arr1[i]]=a1[arr1[i]];}
else
	{a1[arr1[i]]+=(bd2-bd1);}
if(maxv1[i]<=bd1 && machb[i]<=bd1 || maxv1[i]>=bd2 && machb[i]>=bd2)
	{b1[arr1[i]]=b1[arr1[i]];}
else
	{b1[arr1[i]]+=(bd2-bd1);}
if(maxv2[i]<=bd1 && machc[i]<=bd1 || maxv2[i]>=bd2 && machc[i]>=bd2)
	{c1[arr1[i]]=c1[arr1[i]];}
else
	{c1[arr1[i]]+=(bd2-bd1);}}
cout<<"\n\n\nTotal Elapsed Time (T) = "<<machc[n];
int j11[16];
for(i=1;i<=n;i++)
{	j11[i]=i;a11[arr1[i]]=a1[arr1[i]];b11[arr1[i]]=b1[arr1[i]];c11[arr1[i]]=c1[arr1[i]];}
cout<<endl<<"Modified Processing time after breakdown for the machines is:\n";
cout<<"Jobs"<<"\t"<<"Machine	M1"<<"\t"<<"\t"<<"Machine	M2"	<<"\t"<<"\t"<<"Machine
M3"<<"\t"<<"Weightage"<<endl;
for(i=1;i<=n;i++)
{cout<<endl;cout<<j11[i]<<"\t"<<a11[i]<<"\t"<<b11[i]<<"\t"<<c11[i]<<"\t"<<w[i];cout<<endl;}
float maxa12,minb12,minb22,maxc12;float g12[16],h12[16];
//Function for two ficticious machine G and H
//Findinglargest in a11
	maxa12=a11[1]+T12[1]-s22[1];
	for(i=2;i<n;i++)
	{if((a11[i]+T12[i]-s22[i-1])>maxa12)maxa12=a11[i]+T12[i]-s22[i-1];}
	//For finding smallest in b11
	minb12=b11[1]+T23[1]-s33[1];
	for(i=2;i<n;i++)
	{if((b11[i]+T23[i]-s33[i])<minb12)  minb12=b11[i]+T23[i]-s33[i];}
	minb22=b11[1]+T12[1]-s11[i];
	for(i=2;i<n;i++)
	{if((b11[i]+T12[i]-s11[i])<minb22)minb22=b11[i]+T12[i]-s11[i];}
	//Finding largest in c12
	maxc12=c11[1]+T23[1]-s22[1];
	for(i=2;i<n;i++)
	{if((c11[i]+T23[i]-s22[i])>maxc12)maxc12=c11[i]+T23[i]-s22[i];}
if(maxa12>=minb22||maxc12>=minb12)
	{g12[i]=fabs(a11[i]-T12[i]-b11[i]-T23[i]-maxs[i]);h12[i]=fabs(c11[i]-T12[i]-b11[i]-T23[i]+s33[i]);}
else
	{cout<<"\n data is not in Standard Form...\nExitting";getch();exit(0);}
	for(i=1;i<=n;i++)
	{ g12[i]=fabs(a11[i]-T12[i]-b11[i]-T23[i]-maxs[i]); h12[i]=fabs(c11[i]-T12[i]-b11[i]-T23[i]+s33[i]);}
cout<<endl<<"Expected processing time for two fictious machines G and H: \n";
   for(i=1;i<=n;i++)
```

```
  {cout<<endl;cout<<j11[i]<<"\t"<<g12[i]<<"\t"<<h12[i]<<"\t"<<w[i];cout<<endl;}
//To find minimum of G & H
float g11[16],h11[16];
for (i=1;i<=n;i++)
if(g12[i]<=h12[i])
{g11[i]=g12[i]+w[i];h11[i]=h12[i];}
else
{g11[i]=g12[i];h11[i]=h12[i]+w[i];}
float g21[16],h21[16];
for(i=1;i<=n;i++)
{g21[i]=g11[i]/w[i];h21[i]=h11[i]/w[i];}
cout<<endl<<endl<<"displaying original scheduling table"<<endl;
for(i=1;i<=n;i++)
{cout<<j11[i]<<"\t"<<g21[i]<<"\t"<<h21[i]<<endl;}
//calculate G_Beta and H_Beta
if(g21[group[1]]<h21[group[0]])
{minv1=g21[group[1]];}
else
{minv1=h21[group[0]];}
gbeta1=g21[group[0]]+g21[group[1]]-minv1;hbeta1=h21[group[0]]+h21[group[1]]-minv1;
cout<<endl<<endl<<"G_Beta1="<<gbeta1;cout<<endl<<"H_Beta1="<<hbeta1;
int j2[16];float g14[16],h14[16];int f1=1;
for(i=1;i<=n;i++)
{if(j11[i]==group[0]||j11[i]==group[1])
{f1--;}
else
{j2[f1]=j11[i];}f1++;}j2[n-1]=17;
for(i=1;i<=n-2;i++)
{g14[i]=g21[j2[i]];h14[i]=h21[j2[i]];}
g14[n-1]=gbeta1;h14[n-1]=hbeta1;
cout<<endl<<endl<<"displaying original scheduling table"<<endl;
for(i=1;i<=n-1;i++)
{cout<<j2[i]<<"\t"<<g14[i]<<"\t"<<h14[i]<<endl;}
float mingh1[16];char ch1[16];
    for(i=1;i<=n-1;i++)
    {if(g14[i]<h14[i]) {
                mingh1[i]=g14[i];ch1[i]='g';}
else
          {mingh1[i]=h14[i];ch1[i]='h'; }}
          for(i=1;i<=n-1;i++)
     {for(int j=1;j<=n-1;j++)
if(mingh1[i]<mingh1[j])
          {float temp=mingh1[i]; int temp1=j2[i]; char d=ch1[i];mingh1[i]=mingh1[j]; j2[i]=j2[j];
ch1[i]=ch1[j];mingh1[j]=temp; j2[j]=temp1; ch1[j]=d;}}
// calculate beta scheduling
float sbeta1[16];int t2=1,s21=0;
for(i=1;i<=n-1;i++)
     {if(ch1[i]=='h')
     { sbeta1[(n-s21-1)]=j2[i]; s21++;}
else  if(ch1[i]=='g'){sbeta1[t2]=j2[i];t2++;}}
int arr2[16], m1=1;
     cout<<endl<<endl<<"Job Scheduling:"<<"\t";
```

```
for(i=1;i<=n-1;i++)
	{if(sbeta1[i]==17)
	{		arr2[m1]=group[0];arr2[m1+1]=group[1];cout<<group[0]<<"		"<<group[1]<<"
";m1=m1+2;continue;}
else	{cout<<sbeta1[i]<<" ";arr2[m1]=sbeta1[i];m1++;}}
//calculating total computation sequence;
float time1=0.0,macha12[16] ;float maxv11[16],maxv21[16];
	macha1[1]=time1+a11[arr2[1]];
for(i=2;i<=n;i++)
	{macha12[i]=macha1[i-1]+s11[arr2[i-1]];macha1[i]=macha12[i]+a11[arr2[i]];}
	machb1[1]=macha1[1]+b11[arr2[1]]+T12[arr2[1]];
for(i=2;i<=n;i++)
	{if((machb1[i-1]+s22[arr2[i-1]])>(macha1[i]+T12[arr2[i]])){maxv11[i]=machb1[i-1]+s22[arr2[i-1]];}
else {maxv11[i]=macha1[i]+T12[arr2[i]];}machb1[i]=maxv11[i]+b1[arr2[i]];}
	machc1[1]=machb1[1]+c11[arr2[1]]+T23[arr2[1]];
for(i=2;i<=n;i++)
	{if((machc1[i-1]+s33[arr2[i-1]])>(machb1[i]+T23[arr2[i]]))maxv21[i]=machc1[i-1]+s33[arr2[i-1]];
else maxv21[i]=machb1[i]+T23[arr2[i]];machc1[i]=maxv21[i]+c1[arr2[i]];}
float wft,sum1,sum2;sum2=0.0;
for(i=1;i<=n;i++)
{sum2=sum2+w[i];}
///displaying solution
cout<<"\n\n\n\n\n\t\t\t	#####THE SOLUTION##### ";
cout<<"\n\n\t***************************************************************";
cout<<"\n\n\n\t	Optimal Sequence is : ";
for(i=1;i<=n;i++)
	{cout<<" "<<arr2[i];}
cout<<endl<<endl<<"In-Out Table is:"<<endl<<endl;
cout<<"Jobs"<<"\t"<<"Machine M1"<<"\t"<<"\t"<<"Machine M2" <<"\t"<<"\t"<<"Machine M3"<<endl;
cout<<arr2[1]<<"\t"<<time1<<"--"<<macha1[1]<<"
\t"<<"\t"<<macha1[1]+T12[arr2[1]]<<"--"<<machb1[1]<<"
\t"<<"\t"<<machb1[1]+T23[arr2[1]]<<"--"<<machc1[1]<<endl;sum1=0.0;
for(i=2;i<=n;i++)
	{cout<<arr2[i]<<"\t"<<macha12[i]<<"--"<<macha1[i]<<"
"<<"\t"<<maxv11[i]<<"--"<<machb1[i]<<" "<<"\t"<<maxv21[i]<<"--"<<machc1[i]<<endl;
sum1=sum1+(machc1[i]-macha12[i])*w[i];	}
cout<<"\n\n\nTotal Elapsed Time (T) = "<<machc1[n];
wft=((machc1[1]*w[1])+sum1)/sum2;
cout<<"\n\n\n The mean weighted flow time is = "<<wft;
cout<<"\n\n\t***************************************************************";
getch();
}
```

7. Numerical Illustration

Consider 5 jobs, 3 machine flow shop problem with processing time, setup time each associated with their respective probabilities, transportation time and weight of jobs are as given in table.

Jobs	Machine M_1				$T_{i,1\to2}$	Machine M_2				$T_{i,2\to3}$	Machine M_3				Weight of jobs
i	a_{i1}	p_{i1}	s_{i1}	q_{i1}		a_{i2}	p_{i2}	s_{i2}	q_{i2}		a_{i3}	p_{i3}	s_{i3}	q_{i3}	w_i

1	16	0.2	6	0.1	2	4	0.2	7	0.1	2	12	0.1	3	0.2	4
2	12	0.3	7	0.2	1	6	0.2	6	0.3	1	8	0.2	4	0.3	3
3	13	0.2	4	0.3	2	5	0.2	3	0.4	2	15	0.2	6	0.2	2
4	15	0.2	7	0.3	3	4	0.2	3	0.1	3	4	0.2	5	0.1	1
5	14	0.1	4	0.1	4	6	0.2	6	0.1	1	6	0.3	4	0.2	5

Table 2

Our objective is to obtain optimal or near optimal sequence when the break down interval is (a, b) = (10,15) and jobs 2 & 5 are to be processed as equivalent job β = (2,5). Also calculate the total elapsed time and mean weighted flow time.

Solution: As per Step 1: The expected processing times and expected setup times for machines M_1, M_2 and M_3 are as shown in table 3.

As per Step 2: Here, $\text{Max}\{A_{i1} + T_{i,1\to2} - S_{i2}\} \geq \text{Min}\{A_{i2} + T_{i,1\to2} - S_{i1}\}$

$$\text{Max}\{A_{i3} + T_{i,2\to3} - S_{i2}\} \geq \text{Min}\{A_{i2} + T_{i,2\to3} - S_{i3}\}$$; hence feasible solution.

As per Step. 3: The two fictitious machines G and H with processing times G_i and H_i are as shown in table 4.

As per Step 4 &5: The new reduced problem with processing time G_i'' and H_i'' are as shown in table 5

As per step 6: The expected processing time of job block β(2,5) on fictitious machine G & H using equivalent job block criteria given by Maggu & Das (1977) are

$$G''_\beta = 0.466 + 1.08 - 1.08 = 0.466$$
$$H''_\beta = 1.133 + 1.72 - 1.08 = 1.77$$

As per Step 8: The optimal sequence with minimum elapsed time using Johnson's technique is S = β – 4 – 3 – 1 i.e. 2 – 5 – 4 – 3 – 1

As per Step 9 & 10: The In-Out flow table for sequence S is as shown in table 6.

As per Step 11: On considering the effect of the break down interval (10 , 15) the original reduces to as shown in table 7.

Now, On repeating the procedure to get the optimal sequence for the modified scheduling problem , we have the sequence 3 – 2 – 5 – 4 – 1 which is optimal or near optimal. The In-Out flow table for the modified scheduling problem is as shown in table 8.

The mean weighted flow time =

$$\frac{15.6\times3+(18.4-8.8)\times3+(33.2-13.8)\times5+(35.6-15.6)\times1+(37.3-20.7)\times4}{5+3+2+4+1} = 15.53$$

Hence the total elapsed time is 37.3 hrs and the mean weighted flow time is 15.53 hrs.

Conclusion

The new method provides an optimal scheduling sequence with minimum total elapsed time whenever mean weighted production flow time is taken into consideration for 3-machines, n-jobs flow shop scheduling problems. This method is very easy to understand and will help the decision makers in determining a best schedule for a given sets of jobs effectively to control job flow and provide a solution for job sequencing. The study may further be extended by introducing the concept of Rental policy, due date etc.

References

Baker, K. R.(1974), "Introduction of sequencing and scheduling", *John Wiley and Sons*, New York.

Bansal, S. P.(1986), "Resultant job in restricted two machine flow shop problem", *IJOMAS*, 2(1986), 35-45.

Bellman, R.(1956), "Mathematical aspects of scheduling theory", *J. Soc. Indust. Appl. Math.*, 4(1956), 168-205.

Belwal & Mittal (2008), "n jobs machine flow shop scheduling problem with break down of machines, transportation time and equivalent job block", *Bulletin of Pure & Applied Sciences-Mathematics*, source Vol. 27, Source Issue 1.

Chandramouli, A. B.(2005), "Heuristic approach for n-jobs, 3-machines flow-shop scheduling problem involving transportation time, breakdown time and weights of jobs", *Mathematical and Computational Applications*,301-305.

Gupta, D. & Sharma, S. (2011), "Minimizing rental cost under specified rental policy in two stage flow shop , the processing time associated with probabilities including breakdown interval and Job-block criteria", *European Journal of Business and Management*, 3(2),85-103.

Gupta, D., Sharma, S. & Seema (2011), "Heuristic approach for n-jobs, 3-machines flowshop scheduling problem under specified rental policy, processing time associated with probabilities involving transportation time, breakdown interval, Weightage of jobs and job block criteria ", *Mathematical Theory and Modeling*,1(1),30-36.

Johnson, S.M. (1954), "Optimal two and three stage production schedule with set up times included", *Naval Research Logistics Quart.*,1(1), 61-68.

Khodadadi, A.(2008), "Development of a new heuristic for three machines flow-shop scheduling problem with transportation time of jobs", *World Applied Sciences Journal* ,5(5), 598-601

Maggu, P.L. & Das G. (1977), "Equivalent jobs for job block in job scheduling", *Opsearch*, 14(4), 277-281.

Miyazaki, S.& Nishiyama, N.(1980), "Analysis for minimizing weighted mean flow time in flow shop scheduling", *J. O. R. Soc. Of Japan*, 32 (1980), 118-132.

Nawaz, Ensore & Ham (1983), "A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem", *OMEGA*, 91-95.

Narain, L. & Bagga, P.C.(1998), "Two machine flow shop problem with availability constraint on each machine", *JISSOR*, XXIV 1-4, 17-24.

Singh T.P.(1985), "On 2 x n flow-shop problems involving job-block, transportation time, arbitrary time and break down machine time", *PAMS*, XXI, No. 1—2 March.

Pandian, P. &. Rajendran, P. (2010), "Solving Constraint flow shop scheduling problems with three machines", *Int. J. Contemp. Math. Sciences*, 5(19), 921-929.

Yoshida & Hitomi (1979), "Optimal two stage production scheduling with set up times separated", *AIIE Transactions*, II, 261-263

Tables

Table 3: The expected processing times and expected setup times for machines M_1, M_2 and M_3 are

Jobs	A_{i1}	S_{i1}	$T_{i,1\to2}$	A_{i2}	S_{i2}	$T_{i,2\to3}$	A_{i3}	S_{i3}	w_i
1	3.2	0.6	2	0.8	0.7	2	1.2	0.6	4
2	3.6	1.4	1	1.2	1.8	1	1.6	1.2	3
3	2.6	1.2	2	1.0	1.2	2	3.0	1.2	2
4	3.0	2.1	3	0.8	0.3	3	0.8	0.5	1
5	1.4	0.4	4	1.2	0.6	1	1.8	0.8	5

Table 4: The two fictitious machines G and H with processing times G_i and H_i are

Jobs	G_i	H_i	w_i
1	2.3	3	4
2	1.4	0.4	3
3	3.6	0.8	2
4	5.9	5.5	1
5	5.4	3.6	5

Table 5: The new reduced problem with processing time G_i'' and H_i'' are

Jobs	G_i''	H_i''
1	1.575	0.75
2	0.466	1.133
3	1.8	1.4
4	5.9	6.5
5	1.08	1.72

Table 6: The In-Out flow table for sequence S is

Jobs	Machine M_1	$T_{i,1\to2}$	Machine M_2	$T_{i,2\to3}$	Machine M_3	w_i
i	In – Out		In – Out		In - Out	
2	0 – 3.6	1	4.6 – 5.8	1	6.8 – 8.4	3
5	5 – 6.4	4	10.4 – 11.6	1	12.6 – 14.4	5
4	6.8 – 9.8	3	12.8 – 13.6	3	16.6 – 17.4	1
3	11.9 – 14.5	2	16.5 – 17.5	2	19.5 – 22.5	2
1	15.7 – 18.9	2	20.9 – 21.7	2	23.7 – 24.9	4

Table 7: On considering the effect of the break down interval (10, 15) the original reduces to

Jobs	A_{i1}	S_{i1}	$T_{i,1\to2}$	A_{i2}	S_{i2}	$T_{i,2\to3}$	A_{i3}	S_{i3}	w_i
1	3.2	0.6	2	0.8	0.7	2	1.2	0.6	4
2	3.6	1.4	1	1.2	1.8	1	1.6	1.2	3
3	7.6	1.2	2	1	1.2	2	3.0	1.2	2
4	3.0	2.1	3	5.8	0.3	3	0.8	0.5	1
5	1.4	0.4	4	6.2	0.6	1	6.8	0.8	5

Table 8: The In-Out flow table for the modified scheduling problem is

Jobs	Machine M_1	$T_{i,1\to 2}$	Machine M_2	$T_{i,2\to 3}$	Machine M_3	w_i
i	In – Out		In – Out		In - Out	
3	0.0 – 7.6	2	9.6 – 10.6	2	12.6 – 15.6	2
2	8.8 – 12.4	1	13.4 – 14.6	1	16.8 – 18.4	3
5	13.8 – 15.2	4	19.2 – 25.4	1	26.4 – 33.2	5
4	15.6 – 18.6	3	26.0 – 31.8	3	34.8 – 35.6	1
1	20.7 – 23.9	2	32.1 – 32.9	2	36.1 – 37.3	4