European Journal of Business and Management www.iiste.org
ISSN 2222-1905 (Paper) ISSN 2222-2839 (Online)
Vol 3, No.3

Optimal Two Stage Flow Shop Scheduling to Minimize the Rental Cost including Job Block Criteria, Set Up Times and Processing Times Associated with Probabilities

Deepak Gupta
Prof. & Head, Dept. of Mathematics,
Maharishi Markandeshwar University, Mullana, Ambala, India
Tel: +91-9896068604, Email: guptadeepak2003@yahoo.co.in

Naveen Gulati
Assistant Professor, Dept. of Mathematics,
S.D.College, Ambala Cantt , Haryana, India
Email: naveengulatimaths@gmail.com

Sameer Sharma (Corresponding Author)
Assistant Professor, Dept. of Mathematics,
D.A.V. College, Jalandhar, Punjab, India
Tel: +91-9814819064, Email: samsharma31@yahoo.com

Payal Singla
Research Scholar, Dept. of Mathematics
Maharishi Markandeshwar University, Mullana, Ambala, India
Email: payalsingla86@gmail.com

Abstract

This paper is an attempt to study the two stage flow shop scheduling problem in which the processing time and independent set up times of the jobs are associated with probabilities to minimize the rental cost under restrictive rental policy including equivalent job-block criteria. The study gives an optimal schedule rule in order to minimize the rental cost of machines through heuristic approach. The proposed method is very simple and easy to understand and also, provide an important tool for decision makers. To make the method effective and justified a computer program followed by a numerical illustration is given.

Keywords

Johnson's technique, Optimal sequence, Equivalent-job, Flow shop, Rental policy, Makespan, Utilization time, Elapsed time, Idle time.

Mathematical Subject Classification: 90B35, 90B30

1. Introduction

Scheduling is a important process widely used in manufacturing, production, management, computer science and so on. Appropriate scheduling can reduce rental cost of machines and running time of machines. Finding good schedule for given sets of jobs can help factory supervisors effectively to control job flows and provide solutions for job sequencing. A flow shop scheduling problem consists of determining the processing sequence for *n* jobs on *M* machines, where each job is processed on all the machines in the same order and objective is to minimize the time required to process all the jobs. The basic study in flow shop scheduling has been made by Johnson [1954]. The work was developed by Ignall & Scharge[1965], Bagga P.C.[1969], Maggu and Das [1977], Szwarch [1977],Yoshida & Hitomi[1979], Singh T.P.[1985], Chandra Sekhran[1992], Anup[2002], Gupta Deepak[2005] by considering various parameters. Maggu and Das introduced the concept of job-block in the theory of scheduling. This concept is useful and significant in the sense to create a balance between the cost of providing priority in service to the customer and cost of giving services with non-priority customers. The decision maker may decide how much to charge extra to priority customers. Bagga P.C. & Narain studied $n \times 2$ flow shop scheduling problem to minimize rental cost under pre-defined rental policy. Further Singh T.P. and Gupta Deepak [2005], Gupta Deepak and Sharma Sameer [2011] associated probabilities in their studies to minimize the rental cost of machines under predefined rental policies.

In this paper we have extended the study made by Narain, Singh T.P.and Gupta Deepak by introducing the set up time separated from processing time, each associated with probabilities including job block criteria. Here we have developed an algorithm for minimization of utilization of 2 nd machine combined with Johnson's algorithm to solve it. The problem discussed here is wider and has significant use of theoretical results in process industries.

2. Practical Situation

Various practical situations occur in real life when one has got the assignments but does not have one's own machine or does not have enough money or does not want to take risk of investing huge amount of money to purchase machine. Under such circumstances, the machine has to be taken on rent in order to complete the assignments. In his starting career, we find a medical practitioner does not buy expensive machines say X-ray Machine, the Ultra Sound Machine, Rotating Triple Head Single Positron Emission Computed Tomography Scanner, Patient Monitoring Equipment etc. but instead preferred to takes on rent. Renting enables saving working capital, gives option for having the equipment, and allows up gradation to new technology. Sometimes the priority of one job over the other is preferred. It may be because of urgency or demand of its relative importance, the job block criteria becomes important. Moreover in hospitals, industries concern, sometimes the priority of one job over the other is considered. It may be because of urgency or demand of its relative importance. Hence the job block criterion becomes significant.

3. Notations

S : Sequence of jobs 1,2,3,….,n

M_j : Machine j, j= 1,2,…….

$A_{\alpha i}$: Processing time of i^{th} job on machine A.

$B_{\alpha i}$: Processing time of i^{th} job on machine B.

$A'_{\alpha i}$: Expected processing time of i^{th} job on machine A.

$B'_{\alpha i}$: Expected processing time of i^{th} job on machine B.

p_i : Probability associated to the processing time A_i of i^{th} job on machine A.

q_i : Probability associated to the processing time B_i of i^{th} job on machine B.

β : Equivalent job for job – block.

S_i^A : Set up time of i^{th} job on machine A.

S_i^B : Set up time of i^{th} job on machine B.

r_i : Probability associated to the set up time A_i of i^{th} job on machine A.

s_i : Probability associated to the set up time B_i of i^{th} job on machine B.

S_i : Sequence obtained from Johnson's procedure to minimize rental cost.

C_j : Rental cost per unit time of machine j.

U_i : Utilization time of B (2 nd machine) for each sequence S_i

$t_1(S_i)$: Completion time of last job of sequence S_i on machine A.

$t_2(S_i)$: Completion time of last job of sequence S_i on machine B.

$R(S_i)$: Total rental cost for sequence S_i of all machines.

$CT(S_i)$:Completion time of 1^{st} job of each sequence S_i on machine A.

4. Problem Formulation

Let n jobs say $\alpha_1, \alpha_2, \alpha_3 \ldots \alpha_n$ are processed on two machines A & B in the order AB. A job $\alpha_i (i=1,2,3\ldots n)$ has processing time $A_{\alpha i}$ & $B_{\alpha i}$ on each machine respectively with their respective responsibilities p_i & q_i such that $0 \le p_i \le 1$, $\Sigma p_i = 1$, $0 \le q_i \le 1$, $\Sigma q_i = 1$. Let the setup times S_i^A & S_i^B are being separated from processing time associated with respective probabilities r_i & s_i on each machine. Let an equivalent job β is defined as (α_k , α_m) where α_k , α_m are any jobs among the given n jobs such that α_k occurs before job α_m in the order of job block (α_k , α_m). The mathematical model of the problem in matrix form can be stated as in table 1.

Our objective is to find the optimal schedule of all jobs which minimize the utilization time of machines and hence the total rental cost, when costs per unit time for machines A & B are given.

5. Assumptions

1. We assume rental policy that all the machines are taken on rent as and when they are required and are returned as when they are no longer required for processing. Under this policy second machine is taken on rent at time when first job completes its processing on first machine. Therefore idle time of second machine for first job is zero.
2. Jobs are independent to each other.
3. Machine break down is not considered. This simplifies the problem by ignoring the stochastic component of the problem.
4. Pre- emption is not allowed i.e. once a job started on a machine, the process on that machine can't be stopped unless the job is completed.
5. n jobs are processed through two machines A & B in the order AB with processing time $A_{\alpha i}$ & $B_{\alpha i}$ and separate setup time S_i^A & S_i^B respectively.
6. Set up for the processing of a job on machine can be done before completion of the operation of job on A if there exit some ideal time on B.
7. $\Sigma p_i =1, \Sigma q_i =1, \Sigma r_i =1, \Sigma s_i =1$
8. It is given to sequence k jobs $i_1, i_2 \ldots i_k$ as a block or group-job in the order $(i_1, i_2 \ldots i_k)$ showing priority of job i_1 over i_2 etc.

6. Algorithm

To obtain optimal schedule, we proceed as

Step 1. Define expected processing time $A'_{\alpha i}$ & $B'_{\alpha i}$ on machine A & B respectively as follows:

i. $A'_{\alpha i} = A_{\alpha i} \times p_i - S_i^B \times s_i$

ii. $B'_{\alpha i} = B_{\alpha i} \times q_i - S_i^A \times r_i$

Step 2. Take equivalent job $\beta = (\alpha_k, \alpha_m)$ and define processing time as follows:

i. $A'_\beta = A'_{\alpha k} + A'_{\alpha m} - \min(A'_{\alpha m}, B'_{\alpha k})$

ii. $B'_\beta = B'_{\alpha k} + B'_{\alpha m} - \min(A'_{\alpha m}, B'_{\alpha k})$

Step 3. Define a new reduced problem with processing time $A'_{\alpha i}$ & $B'_{\alpha i}$ as define in step 1 and job(α_k , α_m) are replaced by single equivalent job β with processing time A'_β & B'_β as defined in step 2 above.

Step 4. Apply Johnson's (1954) technique to find an optimal schedule of given jobs.

Step 5: Observe the processing time of 1st job of S_1 on the first machine *A*. Let it be α.

Step 6 : Obtain all the jobs having processing time on *A* greater than α. Put these job one by one in the 1 st position of the sequence S_1 in the same order. Let these sequences be $S_2, S_3, S_4 \ldots S_r$.

Step 7 : Prepare in-out table for each sequence $S_i(i = 1,2,\ldots r)$ and evaluate total completion time of last job of each sequence $t_1 (S_i) \,\&\, t_2(S_i)$ on machine *A* & *B* respectively.

Step 8: Evaluate completion time $CT(S_i)$ of 1st job of each sequence S_i on machine *A*.

Step 9: Calculate utilization time U_i of 2nd machine for each sequence S_i as:

$U_i = t_2(S_i) - CT(S_i)$ *for* $i=1,2,3,\ldots r$.

Step 10: Find $Min \{U_i\}$, $i=1, 2\ldots r$. let it be corresponding to $i=m$, then S_m is the optimal sequence for minimum rental cost.

Min rental cost = $t_1 (S_m) \times C_1 + U_m \times C_2$

Where C_1 & C_2 are the rental cost per unit time of 1st & 2 nd machine respectively.

7. Program

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

void main()
{    clrscr();
        float M1[2][20];
        float M2[2][20];
        float sm1[2][20];
        float sm2[2][20];
        int jobs,j,i;
        float check=0;
        cout<<"Enter the number of jobs<=20\n";
        cin>>jobs;
        if(jobs<=1 || jobs>20)
        {
                cout<<"Incorrect Jobs";
                exit(0);
        }
        cout<<"Enter time required for jobs by number 1 Machine";
        for( i=0;i<2;i++)
```

```
{
        if(i==1)
        {    ab:
        check=0;
        i=1;
                cout<<"Enter Probbilities for jobs by machine 1"<<endl;
        }
        for( j=0;j<jobs;j++)
        {
                cin>>M1[i][j];
        }
}
 for(i=0;i<jobs;i++)
 {
 check=check+M1[1][i];
 }
 if(check!=1)
 {
 cout<<"sum of probability should be = 1\n";
 goto ab;
 }
cout<<"Enter time required for jobs by number 2 Machine";
for( i=0;i<2;i++)
{
        if(i==1)
        {
        ab1:
        i=1;
        check=0;
        cout<<"Enter Probbilities for jobs by machine 2"<<endl;
        }
        for( j=0;j<jobs;j++)
        {
                cin>>M2[i][j];
        }
}
  for(i=0;i<jobs;i++)
```

```
 {
 check=check+M2[1][i];
 }
 if(check!=1)
 {
 cout<<"sum of probability should be = 1\n";
 goto ab1;
 }
		cout<<"Enter the   setup time  for jobs by number 1 Machine";
for( i=0;i<2;i++)
{
		if(i==1)
		{
				abc:
				i=1;
				check=0;
				cout<<"Enter Probbilities for setup time  jobs by machine 1"<<endl;
		}
		for( j=0;j<jobs;j++)
		{
				cin>>sm1[i][j];
		}
}
 for(i=0;i<jobs;i++)
 {
 check=check+sm1[1][i];
 }
 if(check!=1)
 {
 cout<<"sum of probability should be = 1\n";
 goto abc;
 }
		cout<<"Enter the   setup time  for jobs by number 2 Machine";
for( i=0;i<2;i++)
{
		if(i==1)
		{
```

```
                abcd:
                i=1;
                check=0;
                cout<<"Enter Probbilities for setup time  jobs by machine 2"<<endl;
            }
            for( j=0;j<jobs;j++)
            {
                cin>>sm2[i][j];
            }
        }
        for(i=0;i<jobs;i++)
        {
        check=check+sm2[1][i];
        }
        if(check!=1)
        {
        cout<<" sum of probability should be =1\n";
        goto abcd;
        }


        int GroupJob[2];
        cout<<"Enter Group Jobs\n";
        for(i=0;i<2;i++)
        {
            cin>>GroupJob[i];
        }
        float cost1,cost2;
        cout<<"Enter cost for machine1 \n";
        cin>>cost1;
        cout<<"Enter cost for machine2 \n";
        cin>>cost2;
        float EPT1[20],EPT2[20];
        for(i=0;i<jobs;i++)
        {
            EPT1[i]=M1[0][i]*M1[1][i]-sm2[0][i]*sm2[1][i];
            cout<<"ept1"<<EPT1[i]<<endl;
```

```
        EPT2[i]=M2[0][i]*M2[1][i]-sm1[0][i]*sm1[1][i];
        cout<<"ept"<<EPT2[i]<<endl;
}
                float EPT11[20],EPT12[20];
for(i=0;i<jobs;i++)
{
        EPT11[i]=M1[0][i]*M1[1][i];
        cout<<"ept1"<<EPT11[i]<<endl;
        EPT12[i]=M2[0][i]*M2[1][i];
        cout<<"ept"<<EPT12[i]<<endl;
}
        float EST1[20],EST2[20];
for(i=0;i<jobs;i++)
{
        EST2[i]=sm2[0][i]*sm2[1][i];
        cout<<"est2"<<EST2[i]<<endl;
        EST1[i]=sm1[0][i]*sm1[1][i];
        cout<<"est1"<<EST1[i]<<endl;
}
float MAD1=0.0,MAD2=0.0;
        for(j=0;j<2;j++)
        {
                MAD1=MAD1+EPT1[GroupJob[j]-1];
                MAD2=MAD2+EPT2[GroupJob[j]-1];
        }
                cout<<MAD1<<" "<<MAD2<<endl;
                if(EPT1[GroupJob[1]-1]>EPT2[GroupJob[0]-1])
                {
                        MAD1=MAD1-EPT2[GroupJob[0]-1];
                        MAD2=MAD2-EPT2[GroupJob[0]-1];
                }
                else
                {
                        MAD1=MAD1-EPT1[GroupJob[1]-1];
                        MAD2=MAD2-EPT1[GroupJob[1]-1];
                }
                        cout<<MAD1<<" "<<MAD2<<endl;
```

```
int count=0;
j=0;
float Reduced1[20],Reduced2[20];
int order[20];
for(i=0;i<jobs;i++)
{
        if(!(GroupJob[0]==i+1 || GroupJob[1]==i+1))
        {
                Reduced1[j]=EPT1[i];
                Reduced2[j]=EPT2[i];
                order[j]=i+1;
                j++;
        }
        else if(count==0)
        {
                Reduced1[j]=MAD1;
                Reduced2[j]=MAD2;
                order[j]=-1;
                count++;
                j++;
        }
}
cout<<"Reduced1 : "<<endl;;
for(i=0;i<jobs-1;i++)
{
        cout<<Reduced1[i]<<" ";
}
cout<<"Reduced2 : "<<endl;;
for(i=0;i<jobs-1;i++)
{
        cout<<Reduced2[i]<<" ";
}
float min1[20],min2[20],order1[20],order2[20];
j=0;
int k=0;
for(i=0;i<jobs-1;i++)
{
```

```
        if(Reduced1[i]>Reduced2[i])
        {
                min2[j]=Reduced2[i];
                order2[j]=order[i];
                j++;
        }
        else
        {
                min1[k]=Reduced1[i];
                order1[k]=order[i];
                k++;
        }
}
int l,m;
cout<<"min1 is : "<<endl;
for(l=0;l<k;l++)
{
        cout<<min1[l]<<" ";
}
cout<<"min2 is : "<<endl;
for(l=0;l<k;l++)
{
        cout<<min2[l]<<" ";
}
float temp;
int tmp;
for(l=0;l<k-1;l++)
{
        for(m=0;m<k-1;m++)
        {
                if(min1[m]>min1[m+1])
                {
                        temp=min1[m];
                        min1[m]=min1[m+1];
                        min1[m+1]=temp;
                        tmp=order1[m];
                        order1[m]=order1[m+1];
```

```
                order1[m+1]=tmp;
            }
        }
    }
    for(l=0;l<j-1;l++)
    {
        for(m=0;m<j-1;m++)
        {
            if(min2[m]<min2[m+1])
            {
                temp=min2[m];
                min2[m]=min2[m+1];
                min2[m+1]=temp;
                tmp=order2[m];
                order2[m]=order2[m+1];
                order2[m+1]=tmp;
            }
        }
    }
    int real[20];
    m=0;
    cout<<"So the Required sequence is: "<<endl;
    for(i=0;i<k;i++)
    {
        if(order1[i]==-1)
        {
            for(l=0;l<2;l++)
            {
                real[m]=GroupJob[l];
                m++;
                cout<<GroupJob[l]<<endl;
            }
        }
        else
        {
            real[m]=order1[i];
            m++;
```

```
                cout<<order1[i]<<endl;
            }
    }
    for(i=0;i<j;i++)
    {
        if(order2[i]==-1)
        {
                for(l=0;l<2;l++)
                {
                        real[m]=GroupJob[l];
                        m++;
                        cout<<GroupJob[l]<<endl;
                }
        }
        else
        {
                real[m]=order2[i];
                m++;
                cout<<order2[i]<<endl;
        }
    }
    cout<<"Flow time for machine 1"<<endl;
    float time=0.0,time2,initial2;
    for(i=0;i<jobs;i++)
    {

        cout<<real[i]<<"    "<<time<<"  to  "<<time+EPT11[real[i]-1]<<endl;
        if(i==jobs-1)
        {
        initial2=time+EPT11[real[i]-1];
        }
        time=time+EPT11[real[i]-1]+EST1[real[i]-1];

    }
    time=EPT11[real[0]-1];
    cout<<"Flow time for machine 2"<<endl;
    time2=EPT11[real[0]-1];
```

```
float initial;
float last;
initial=time2;
for(i=0;i<jobs;i++)
{

  cout<<real[i]<<"   "<<time2<<"  to  "<<time2+EPT12[real[i]-1]<<endl;
        if(i==jobs-1)
  {
  last=time2+EPT12[real[i]-1];
  }
  time=time+EPT11[real[i+1]-1]+EST1[real[i]-1];
  time2=time2+EPT12[real[i]-1]+EST2[real[i]-1];
  if(time2<time)
  {
  time2=time;
  }

}
cout<<  "total  rental  cost  is  ("<<cost1<<"*"<<initial2<<")+("<<cost2<<"*("<<last<<"-
"<<initial<<")) is" <<(cost1*initial2)+(cost2*(last-initial));
cout<<"other sequences are: \n";
int real1[20];
for(j=0;j<jobs;j++)
{
        if(EPT1[real[0]-1]<EPT1[j])
        {
                for(k=0;k<jobs;k++)
                {
                        if(j+1==real[k])
                        {
                             break;
                        }
                }
                if(j+1== GroupJob[0] ||  j+1== GroupJob[1]  )
                {
                        if( EPT1[GroupJob[0]-1]<=EPT1[real[0]-1])
                        {
```

```
                break;
                }
                real1[0]=GroupJob[0];
                real1[1]=GroupJob[1];
                for(l=k+1;l>1;l--)
                {
                        real1[l]=real[l-2];
                }
                for(l=k+2;l<jobs;l++)
                {
                        real1[l]=real[l];
                }
        }
        else
        {
                real1[0]=j+1;

                for(l=k;l>0;l--)
                {
                        real1[l]=real[l-1];
                }
                for(l=k+1;l<jobs;l++)
                {
                        real1[l]=real[l];
                }
        }
        for(i=0;i<jobs;i++)
        {
                cout<<real1[i]<<" ";
        }
        cout<<endl;
        cout<<"Flow time for machine 1"<<endl;
float time=0.0,time2,initial1;
for(i=0;i<jobs;i++)
{
    cout<<real1[i]<<"   "<<time<<"  to  "<<time+EPT11[real1[i]-1]<<endl;
    if(i==jobs-1)
```

```
                {
                initial1=time+EPT11[real[jobs-1]-1];
                }
                time=time+EPT11[real1[i]-1]+EST1[real1[i]-1];
            }
            time=EPT11[real1[0]-1];
            cout<<"Flow time for machine 2"<<endl;
            time2=EPT11[real1[0]-1];
            float ini;
            ini=time2;
            for(i=0;i<jobs;i++)
            {
             cout<<real1[i]<<"    "<<time2<<"  to  "<<time2+EPT12[real1[i]-1]<<endl;
             if(i==jobs-1)
             {
             last=time2+EPT12[real[i]-1];
             }
             time=time+EPT11[real1[i+1]-1]+EST1[real1[i]-1];
             time2=time2+EPT12[real1[i]-1]+EST2[real1[i]-1];
             if(time2<time)
             {
             time2=time;
             }
            }
                 cout<<"total rental cost is : ("<<cost1<<"*"<<initial1<<")+("<<cost2<<"*("<<last<<"-
"<<ini<<")) "<<(initial1*cost1)+((last-ini)*cost2);
                  getch();
                }
                cout<<endl;
            }
        getch();
}
```

8. Numerical Illustration

Consider 5 jobs and 2 machines problem to minimize the rental cost. The processing and setup times with their respective associated probabilities are given as follows. Obtain the optimal sequence of jobs and minimum rental cost of the complete set up, given rental costs per unit time for machines M_1 & M_2 are 15 and 13 units respectively, and jobs (2, 5) are to be processed as an equivalent group job.

Job	Machine A	Machine B

i	A_i	p_i	S_i^A	r_i	B_i	q_i	S_i^B	s_i
1	16	0.3	8	0.1	15	0.3	13	0.3
2	13	0.2	12	0.2	17	0.2	11	0.2
3	12	0.1	14	0.3	14	0.2	9	0.1
4	15	0.3	17	0.2	18	0.2	21	0.1
5	28	0.1	18	0.2	12	0.1	17	0.3

Solution

Step 1**:** The expected processing times $A_i^{'}$ and $B_i^{'}$ on machine A and B are as in table 2.

Step 2: The processing times of equivalent job block β = (2,5) by using *Maggu* and *Das* criteria are **(show in table 3)** given by

$A_\beta^{'}$ = 0.4 – 2.3 +2.3 = 0.4

And $B_\beta^{'}$ = 1.0 - 0.4 + 2.3 = 0.9

Step 3**:** Using *Johnson's* two machines algorithm, the optimal sequence is

S_1 = β, 1, 4, 3 .i.e. S_1 = 2 – 5 – 1 – 4 – 3 .

Step 4 : The other optimal sequences for minimizing rental cost are

S_2 =1-2-5-4-3, S_3 = 4-2-5-1-3.

Step 5 : The in-out flow tables for sequences S_1, S_2 and S_3 having job block (2, 5) are as shown in **table 4, 5 and 6**.

For S_1= 2 – 5 – 1 – 4 – 3

Total time elapsed on machine A = $t_1(S_1)$ = 26.1

Total time elapsed on machine B = $t_2(S_1)$ = 33.1

Utilization time of 2nd machine (B) = U_1 = 33.1- 2.6= 30.5.

For S_2 = 1 – 2 – 5 – 4 – 3

Total time elapsed on machine A = $t_1(S_3)$ = 26.1

Total time elapsed on machine B = $t_2(S_3)$ = 33.6

Utilization time of 2nd machine (B) = U_2 = 33.6 – 4.8 = 28.8.

For S_3 = 4 –2 – 5 – 1 – 3

Total time elapsed on machine A = $t_1(S_4)$ = 26.1

Total time elapsed on machine B = $t_2(S_4)$ = 35..3

Utilization time of 2nd machine (B) = U_3 = 35.3 – 4.5 = 30.8

The total utilization of A machine is fixed 26.1 units and minimum utilization time of B machine is 28.8 units for the sequence **S_2 =1-2-5-4-3**

Therefore optimal sequence is **S_2 =1-2-5-4-3** and the total rental cost =26.1×15 + 28.8×13 = 765.9 units**.**

References

Johnson S. M. (1954), Optimal two and three stage production schedule with set up times included. Nay Res Log Quart Vol 1, pp 61-68.

Ignall E. and Schrage L. (1965)**,** Application of the branch and bound technique to some flow shop scheduling problems, Operation Research, 13, pp 400-412.

P.C.Bagga (1969), Sequencing in a rental situations, Journal of Canadian Operation Research Society 7 ,pp 152-153

Maggu P. L and Das G. (1977), Equivalent jobs for job block in job sequencing, Opsearch, Vol 14, No. 4, pp 277-281.

Szware W. (1977), Special cases of the flow shop problems, Naval Research Log, Quartly 24, pp 403-492.

Yoshida and Hitomi (1979), O*ptimal* two stage production scheduling with set up times separated., AIIE Transactions, Vol. II, pp 261-263.

Singh T.P (1985), on n×2 shop problem involving job block. Transportation times and Break-down Machine times, PAMS, Vol. XXI, pp 1-2

Chander Sekharan, K. Rajendra, Deepak Chanderi (1992), An Efficient Heuristic Approach to the scheduling of jobs in a flow shop, European Journal of Operation Research 61, pp 318-325.

Anup (2002), On two machine flow shop problem in which processing time assumes probabilities and there exists equivalent for an ordered job block., JISSO Vol XXIII No. 1-4, pp 41-44.

Singh T. P., K. Rajindra & Gupta Deepak (2005), Optimal three stage production schedule the processing time and set times associated with probabilities including job block criteria, Proceedings of National Conference FACM-2005,pp 463-492.

Narian L & Bagga P.C. (2005), Scheduling problems in Rental Situation, Bulletin of Pure and Applied Sciences: Section E. Mathematics and Statistics,Vol.24, ISSN: 0970-6577.

Singh, T.P, Gupta Deepak (2006), Minimizing rental cost in two stage flow shop , the processing time associated with probabilities including job block, Reflections de ERA, Vol 1. Issue 2, pp 107-120.

Gupta Deepak & Sharma Sameer (2011), Minimizing Rental Cost under Specified Rental Policy in Two Stage Flow Shop, the Processing Time Associated with Probabilities Including Break-down Interval and Job – Block Criteria , European Journal of Business and Management Vol 3, No 2, pp 85-103.

Remarks

i. If set up times of each machine is negligible small, the results are similar as Anup [2002].

ii. If probabilities are not associated in the problem, the results tally with Singh T.P.[1998].

iii. The study may be extended further for three machines flow shop, also by considering various parameters such as transportation time, break down interval etc.

Table 1: The mathematical model of the problem in matrix form

Jobs	Machine A				Machine B			
i	A_i	P_i	S_i	r_i	B_i	q_i	S_i	s_i

α_1 α_2 α_3 α_4 --- --- α_n	A_1 A_2 A_3 A_4 --- --- A_n	p_1 p_2 p_3 p_4 --- --- p_n	S_{1A} S_{2A} S_{3A} S_4 ----- --- S_{nA}	r_1 r_2 r_3 r_4 ---- --- r_n	B_1 B_2 B_3 B_4 --- --- B_n	q_1 q_2 q_3 q_4 --- --- q_n	S_{1B} S_{2B} S_{3B} S_4 --- --- S_{nB}	s_1 s_2 s_3 s_4 --- --- s_n

Table 2: The expected processing times $A'_{\alpha i}$ and $B'_{\alpha i}$ on machine A and B are

i	$A'_{\alpha i}$	$B'_{\alpha i}$
1	4.8 – 3.9=0.9	4.5 – 0.8=3.7
2	2.6 – 2.2=0.4	3.4 – 2.4=1.0
3	1.2 – 0.9=0.3	2.8 – 4.2 = -1.4
4	4.5 – 2.1=2.4	3.6 – 3.4=0.2
5	2.8 – 5.1=-2.3	1.2 – 3.6= -2.4

Table 3: The processing times of equivalent job block β = (2, 5) by using *Maggu* and *Das criteria* is

Job	Machine A	Machine B
i	In – Out	In - Out
4	0 – 1.8	1.8 – 3.5
3	2.8 - 4.0	5.1 – 6.7
1	5.2 - 8.5	8.5 – 11.2
2	10.6 - 12.8	12.8 – 14.9
5	14.6 - 16.1	16.4 -18.7

Table 4: The in-out flow table for the sequence S_1 = 2 – 5 – 1 – 4 – 3 is

Job	Machine A	Machine B

i	In – Out	In - Out
2	0 – 2.6	2.6 – 6.0
5	5.0 - 7.8	8.2 – 9.4
1	11.4 – 16.2	16.2 – 20.7
4	17.0 – 21.5	24.6 – 28.2
3	24.9 – 26.1	30.3 – 33.1

Table 5: The in-out flow table for the sequence S_2 = 1 – 2 – 5 – 4 – 3 is

Job	Machine A	Machine B
I	In – Out	In - Out
1	0 – 4.8	4.8 – 9.3
2	5.6 – 8.2	13.2 – 16.6
5	10.6 – 13.4	18.8 - 20.0
4	17.0 – 21.5	25.1 – 28.7
3	24.9 – 26.1	30.8 – 33.6

Table 6: The in-out flow table for the sequence S_3 = 4 –2 – 5 – 1 – 3 is

Job	Machine A	Machine B
i	In – Out	In - Out
4	0 – 4.5	4.5 – 8.1
2	7.9 – 10.5	10.5 – 13.9
5	12.9 – 15.7	16.1 – 17.3
1	19.3 – 24.1	24.1 – 28.6
3	24.9 – 26.1	32.5 – 35.3