Multi Process Three Dimensional Job Assignment Model

P.Guravaraju[1], C. Suresh Babu[2] R. Vijaya Lakshmi[3], M. Sundara Murthy[4]

2-Asst.Professor, sidharth institute of engineering and technology, puttur, Andhra Pradesh, India
1, 3 -Research Scholars, Dept of Mathematics, S.V.University, Tirupati, Chittoor Dt, Andhra Pradesh, India
4- Professor (Rtd), Dept of Mathematics, S.V.University, Tirupati, Chittoor Dt, Andhra Pradesh, India

Abstract:

So many researchers introduced variant problems for assignment model. In this Problem there is a set of n jobs, m machines and r periods. This is three dimensional problem. Number of jobs to be completed is n_0. The total number of assigned jobs should be 2 n_0. D(i,j,k) be the cost array of assignment of i^{th} Job on j^{th} machine at k^{th} period. Each job requires two processes among A,B and C Processes and there are given. Any machine can do process A in 1^{st} or 3^{rd} period, process B in 2^{nd} or 4^{th} period and process C in any one of the four periods. For our convenience we consider processes A, B and C .as 1,2 and 3. One machine can do a job process in one period .i.e different jobs processes can be done on a machine in different periods. Each n_0 job has to be assigned twice to machines for the two processes, hence the total number of assignment is 2 n_0. The objective of the problem is to assign each n_0 job twice to the machines such that the total processing cost is minimum subject to conditions.

Keywords: *assignment, jobs ,machines , periods, process and n_0*

1. INTRODUCTION:

An assignment problem is a special type of linear programming problem where the objective is to minimize the cost or time of completing a number of jobs by a number of persons. One of the important characteristics of assignment problem is that only one job (or worker) is assigned to one machine (or project). This method was developed by D. Konig, a Hungarian mathematician and is therefore known as the Hungarian method of assignment problem. In order to use this method, one needs to know only the cost of making all the possible assignments. Each assignment problem has a matrix (table) associated with it. Normally, the objects (or people) one wishes to assign are expressed in rows, whereas the columns represent the tasks (or things) assigned to them

Recently many researchers introduced various Multi-dimensional Assignment Problems. Assignment is a technique to find out least possible cost by assigning least costs to jobs and workers.
Generally assignment is used in one-one basis which is called Two-dimensional Assignment Problem (ASP) It involves assignment of jobs to machines, workers to jobs and teachers to classes etc.with total minimum cost.

ASP is a special type of linear programming problem and objective of ASP is to minimize the cost and maximize the profit. The Multi-Dimensional Assignment Problem (MAP) is simply a higher dimensional version of linear ASP. The dimension of ASP represents the number of sets of elements are matched.
The term 'multi dimensional' refers to the assignment problems of dimension three or more.

2. VARIATIONS OF ASSIGNMENT PROBLEM:

Sundara murthy. [12]. Suresh Babu. et al [13], and Purushotham [8] had studied the problem called “Pattern Recognition technique based Lexi-Search Approach to the Variant Multi-Dimensional Assignment Problem”. The Multi Dimensional Assignment Problem is a combinational optimization problem that is known to be NP-Hard. Sobhan Babu [10] had also studied on the assignment is called “A new Approach for Variant Multi Assignment Problem”. Vidhyullatha [14] had studied the problem called “Three Dimensional Group Assignment Problem”.

The Hungarian method of Khun et al [7], Labeling process and line covering method are widely used to solve the Assignment problem. Barr et al [2] have proposed alternating basis algorithm, while Hung et al [6] proposed a row algorithm based on resembling the Hungarian Method in some ways but differs substantially in other aspects, where in addition to the cost of workers performing the jobs, a supervisory cost is also considered. A variation of the assignment problem by Geetha et al [5]. The time minimization Assignment Problem (TMAP) is another important class of assignment problem. TMAP has been considered by many researchers like Garfinke [4]. Ross et al [9]. A Branch and Bound Algorithm for Generalized Assignment Problem. Bertsekas [3],. Some special cases of Assignment Problem by Subramanyam [11] and Balakrishna [1] under the usual assumption that work on all the n jobs starts simultaneously.

In our problem there are three dimensions. They are set of jobs, set of machines and set of periods with different job processes called 'Multi

Process Three Dimensional Job Assignment Model'. It varies from other assignment problems

3. PROBLEM DESCRIPTION:

In this problem there is a set of n jobs, J={1,2,….n}, a set of m machines, M={1,2,…m} and a set of r periods, P={1,2,….r}. The number of jobs to be completed is n_0 (n_0<n) i.e the number of assignment of jobs n should be truncated. D(i,j,k) is the cost array of assignment of i^{th} job on j^{th} machine at k^{th} period.

There are three types of job processing A,B and C. Each job requires two types of processing among three processes.The two required job processes for each job (JP) is given.The two processes i.e., JP(i,1), JP(i,2) are the required processes.A machine can do process A in 1^{st} or 3^{rd} period. process B in 2^{nd} or 4^{th} period and process C in anyone of the four periods. Here we identify A process as 1, B process as 2 and C process as 3. The value of r is taken as 4.i.e.r = 4

One job process is done on a machine in one period and the same machine is available in another period for another job.i.e., different jobs can be done on a machine in different periods. Here period is third dimension. Each n_0 job has to be assigned twice to machines for the two processes.

The objective of our problem is to assign each n_0 job twice to the machines such that the total processing cost is minimum subject to conditions.

We presented the Pattern Recognition Technique using Lexi Search Algorithm (LSA) for this model. We tested the proposed algorithm by different set of problems.

4. MATHEMATICAL FORMULATION:

$$\left.\begin{array}{l} \operatorname{Min} Z(X) = \sum_i \sum_j \sum_k D(i,j,k)\, X(i,j,k) \\ i \in J, j \in M, k \in P \quad \ldots\ldots\ldots\ldots (1) \end{array}\right\}$$

$$\left.\begin{array}{l} \sum_i \sum_j \sum_k x(i,j,k) = 2n_0 \\ i \in J, j \in M, k \in P \quad \ldots\ldots (2) \end{array}\right\}$$

$$x(i_1, j_1, k_1) = 1 = x(i_2, j_2, k_2)$$
$$\text{if } j_1 = j_2, \quad k_1 \neq k_2 \quad \ldots\ldots\ldots (3)$$

$$\sum_j \sum_k X(i,j,k) = 2, \forall i \in j$$
$$\ldots\ldots\ldots (4)$$
$$j \in M, k \in P$$

$$x(i,j,k) = 1, JX(i) = 1 \text{ and } \sum_{i=1}^{n} JX(i) = n_0$$
$$\ldots\ldots\ldots (5)$$

$$\left.\begin{array}{l} \text{If } x(i,j,k) = 1, JP(s,i) = l \text{ where } s = 1,2 \\ \text{Then } l = 1 \text{ for } k = 1 \text{ or } 3 \\ \qquad l = 2 \text{ for } k = 2 \text{ or } 4 \\ \qquad l = 3 \text{ for } k = 1,2,3 \text{ or } 4 \end{array}\right\}$$
$$\ldots\ldots\ldots (6)$$

$$x(i,j,k) = 0 \text{ or } 1 \quad \ldots\ldots\ldots\ldots (7)$$

Constraint (1) is the objective function which measures the total processing cost is minimum for all n_0 jobs under the given restrictions

Constraint (2) describes that all the n_0 jobs are assigned twice under the given conditions

Constraint (3) describes the restriction that each machine can do one job process in one period

Constraint (4) describe that each job is processed twice as per its requirement on the machines

Constraint (5) describe that the number of assigned jobs is equal to n_0

Constraint (6) the machine can do processes ‘1’ in the period 1 (or) 3Process ‘2’ in the period 2 or 4 and Process ‘3’ in any period i.e 1or 2or 3 or 4

Constraint (7) describes that the i^{th} job is done on j^{th} machine in k^{th} period then X(i,j,k)=1 other wise zero

5. NUMERICAL ILLUSTRATION:

The concepts and algorithm developed by illustrating a numerical example for which the number of jobs n=8, number of machines m=5 and number of periods r = 4 .Here each n_0 job can be processed twice on machines in different periods .For our convenient the job processes done by machines are identified as 1,2 and 3. Out of 8 jobs n=8 and 5 jobs are to be completed.i.e.n_0 =5. Therefore the total number of assigned processes are 2 n_0=2x5=10.

D (i, j, k) means the cost array of i^{th} job on j^{th} machine at k^{th} period. The tables 1-4 represents the requirement of the cost for doing the job with respect to corresponding machine at particular period. Then the cost array D(i, j, k) is given below.

Table - 1

D (i,j,1)					
	1	2	3	4	5
1	10	23	19	39	22
2	4	11	8	31	–
3	60	47	–	14	54
4	32	3	21	53	–
5	18	37	–	17	59
6	13	43	7	29	23
7	–	28	–	46	7
8	30	–	24	20	5

Table - 2

D (i,j,2)					
	1	2	3	4	5
1	9	34	26	23	–
2	12	28	35	1	15
3	44	10	15	8	–
4	17	32	–	35	45
5	–	16	2	20	14
6	14	5	8	11	–
7	24	27	4	52	58
8	–	18	36	–	22

Table – 3

D (i,j,3)					
	1	2	3	4	5
1	19	–	28	–	50
2	12	27	42	33	–
3	17	21	–	3	55
4	265	23	48	–	9
5	42	38	26	11	6
6	1	–	60	16	–
7	24	13	30	13	45
8	354	–	21	10	25

Table-6

PROCESS	PERIOD
A = 1	1 or 3
B = 2	2 or 4
C = 3	1,2,3 or 4

Table - 4

D(i,j,4)					
	1	2	3	4	5
1	24	40	6	31	34
2	18	–	56	25	–
3	51	19	3	49	21
4	–	57	–	12	2
5	14	15	7	50	8
6	41	24	–	46	–
7	20	–	53	29	32
8	9	1	–	6	18

In the above **tables-1,2**,3 and 4, D(6, 4, 1) =29 means that the cost of the 6th job on 4th machine in1st period is 29. For the table-1,2,3 and 4 we developed Alphabet Table and Search table. . The total number of processes to be assigned is $2n_0$.The objective of the problem is the total assigned job processing cost should be minimum.

Table-5

JP =

JOB	PROCESS	
	I	II
1	2	3
2	2	3
3	1	2
4	2	3
5	1	2
6	2	3
7	1	3
8	1	2

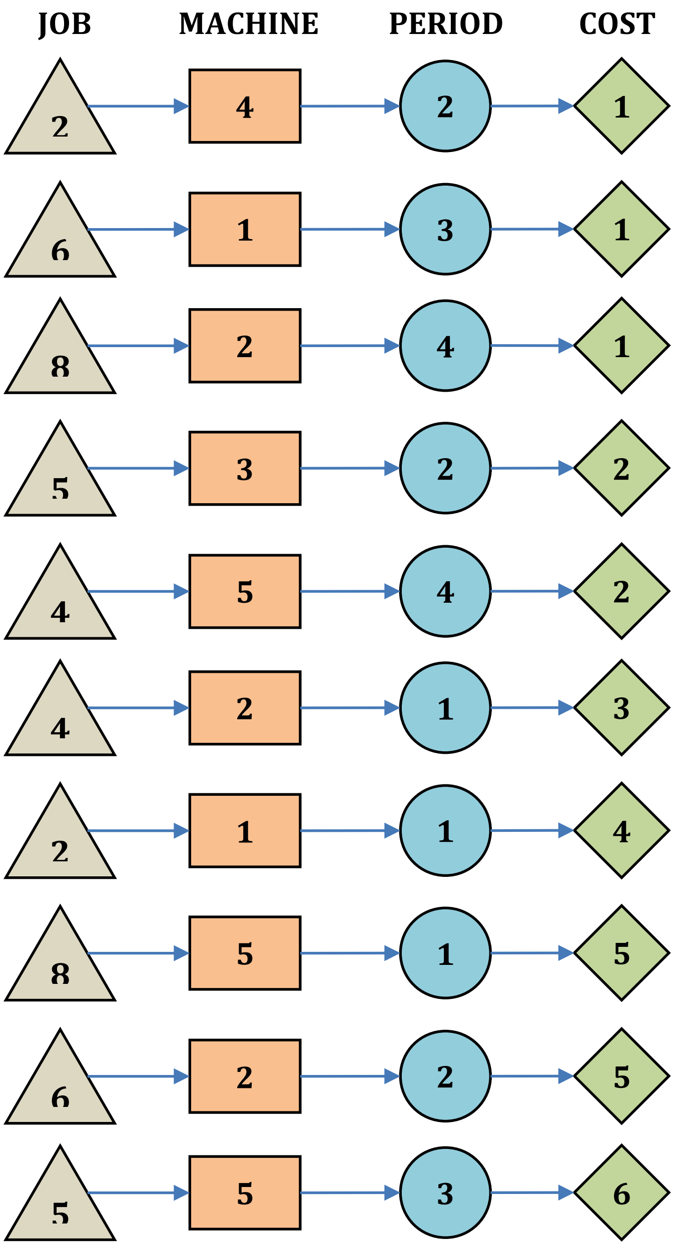
Figure-1

From the Table-5, JP(6,2)=3 i.e.the second process of 6th job is 3 i.e. process C and from the Table-6,process B of a job can be done in 2nd or 4th period on any machine.i.e. process 2 can be done either in second period or fourth period on any machine.

6. *FEASIBLE SOLUTION:*

Consider an ordered triple set (2,4,2),(6,1,3),(8,2,4),(5,3,2),(4,5,4),(4,2,1),(2,1,1),(8,5,1),(6,2,2),(5,5,3) represents a feasible solution for the above numerical example.

The following **figure-1** represents a feasible solution. The triangle shapes represent jobs, The rectangle shapes represent machines, circle shapes represent periods and rhombus shape represents cost. The values in triangle indicates name of the job, The values in rectangle indicate name of the machine, values in circles indicate name of the period and values in rhombus represent corresponding cost of a job on a machine at a period.



The above figure-1 satisfies all the constraints in the Mathematical Formulation. It is a feasible solution. The cost of corresponding ordered triples are:

D(2,4,2)=1,D(6,1,3)=1,D(8,2,4)=1,D(5,3,2)=2,D(4,5,4)=2,D(4,2,1)=3,D(2,1,1)=4,D(8,5,1)=5, D(6,2,2)=5,D(5,5,3)=6.

The total cost=1+1+1+2+2+3+4+5+5+6=30.

7. SOLUTION PROCEDURE:

In the above fig-1 for the feasible solution we observe that there are 10 ordered triples taken along with the values from the cost matrices for the numerical example in tables1-4. The 10 ordered triples are selected such that they represent a feasible solution in fig-1. So the problem is that we have to select 10 ordered triples from the cost matrces (8×5×4) along with values such that the total cost is minimum and represents a feasible solution. For this selection of 10 ordered triples we arrange the 8×5×4=160 ordered triples with the increasing order and call this formation as alphabet table and we will develop an algorithm for the selection along with the checking for the feasibility.

7.1. DEFINITION OF A PATTERN:

An indicator three-dimensional array which is associated with connection is called a 'pattern'. A Pattern is said to be feasible if X is a solution.

$$V(X) = \sum_{i\in J}\sum_{j\in M}\sum_{k\in P} D(i,j,k)X(i,j,k)$$

The value V(x) gives the total cost of the process for the solution represented by X. The pattern represented in the tables 7-10 is a feasible pattern. In the algorithm, which is developed in the sequel, a search is made for a feasible pattern with the least value. Each pattern of the solution X is represented by the set of ordered triple (i, j, k) for which X (i, j, k)=1, with understanding that the other X(i, j, k)'s are zeros.

The ordered triple set {(2,4,2), (6,1,3), (8,2,4), (5,3,2), (4,5,4), (4,2,1), (2,1,1), (8,5,1),(6,2,2), (5,5,3)} represents the pattern as given in tables 7-10, which is the feasible solution.

Table-7

$$\mathbf{X\ (i,j,1)} = \begin{bmatrix} 0&0&0&0&0\\ 1&0&0&0&0\\ 0&0&0&0&0\\ 0&1&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&1 \end{bmatrix}$$

Table-8

$$X(i,j,2) = \begin{bmatrix} 0&0&0&0&0\\ 0&0&0&1&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&1&0&0\\ 0&1&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0 \end{bmatrix}$$

Table-9

$$X(i,j,3) = \begin{bmatrix} 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&1\\ 1&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0 \end{bmatrix}$$

Table-10

$$X(i,j,4) = \begin{bmatrix} 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&1\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&0\\ 0&1&0&0&0 \end{bmatrix}$$

Hence the solution of the above pattern X as follows:

(2,4,2)+(6,1,3)+(8,2,4)+(5,3,2)+(4,5,4)+(4,2,1)+(2,1,1)+(8,5,1)+(6,2,2)+(5,5,3)
=1+1+1+2+2+3+4+5+5+6=30.

.

7.2. ALPHABET TABLE:

There is p×q×r ordered triples in the three-dimensional array X. For convenience these are arranged in ascending order of their corresponding costs and are indexed from 1 to p×q×r (Sundara Murthy-1979). Let SN= [1, 2, 3…] be the set of indices. Let D be the corresponding array of costs. If a, b ∈ SN and a<b then D (a) ≤ D(b). Also let the arrays R, C, P be the array of indices of the ordered triples represented by J ,M and P. CD is the array of cumulative sum of the elements of D. The arrays SN, D, CD, R, C, and P for the numerical example are given in the table-11. If p ∈ SN then (R(p),C(p),P(p)) is the ordered triple and D(a)=D(R(a),C(a),P(a)) is the value of the ordered triple and CD (a) = $\sum_{i=1}^{a} D(i)$

Table-11

ALPHABET TABLE

S.No	D	CD	R	C	P
1	1	1	2	4	2
2	1	2	6	1	3
3	1	3	8	2	4
4	2	5	5	3	2
5	2	7	4	5	4
6	3	10	4	2	1
7	3	13	3	4	3
8	3	16	3	3	4
9	4	20	2	1	1
10	4	24	7	3	2
11	5	29	8	5	1
12	5	34	6	2	2
13	6	4	5	5	3
14	6	46	1	3	4
15	6	52	8	4	4
16	*7*	*59*	*6*	*3*	*1*
17	7	66	7	5	1
18	7	73	5	3	4
19	8	81	2	3	1
20	8	89	3	4	2
21	8	97	6	3	2
22	8	105	5	5	4
23	9	114	1	1	2
24	9	123	4	5	3
25	9	132	8	1	4
26	10	142	1	1	1
27	10	152	3	2	2
28	10	162	8	4	3
29	11	173	2	2	1
30	11	184	6	4	2
31	11	195	5	4	3
32	12	207	2	1	2
33	12	219	2	1	3
34	12	231	4	4	4
35	13	244	6	1	1
36	13	257	7	2	3
37	13	270	7	4	3
38	14	284	3	4	1
39	14	298	5	5	2
40	14	312	6	1	2
41	14	326	5	1	4
42	15	341	2	5	2
43	15	356	3	3	2
44	15	371	5	2	4
45	16	387	5	2	2
46	16	403	6	4	3
47	17	420	5	4	1
48	17	437	4	1	2
49	17	454	3	1	3
50	18	472	5	1	1
51	18	490	8	2	2
52	18	508	2	1	4
53	18	526	8	5	4
54	19	545	1	3	1
55	19	564	1	1	3
56	19	583	3	2	4
57	20	603	8	4	1
58	20	623	5	4	2
59	20	643	7	1	4
60	21	664	3	3	1
61	21	685	4	2	3
62	21	706	8	3	3
63	21	727	3	5	4
64	22	749	1	5	1
65	22	771	8	5	2
66	23	794	1	2	1
67	23	817	6	5	1
68	23	840	1	4	2
69	23	863	4	2	3
70	24	887	8	3	1
71	24	911	7	1	2
72	24	935	7	1	3
73	24	959	1	1	4
74	24	983	6	2	4
75	25	1008	8	5	3
76	25	1033	2	4	4
77	26	1059	1	3	2
78	26	1085	4	1	3
79	26	1111	5	3	3
80	27	1138	7	2	2
81	27	1165	2	2	3
82	28	1193	7	2	1
83	28	1221	2	2	2
84	28	1249	1	3	3
85	29	1278	6	4	1
86	29	1307	7	4	4

87	30	1337	8	1	1
88	30	1367	7	3	3
89	31	1398	2	4	1
90	31	1429	1	4	4
91	32	1461	4	1	1
92	32	1493	4	2	2
93	32	1525	7	5	4
94	33	1558	2	4	3
95	34	1592	1	2	2
96	34	1626	1	5	4
97	35	1661	2	3	2
98	35	1696	4	4	2
99	36	1732	8	3	2
100	36	1768	8	1	3
101	37	1805	5	2	1
102	38	1843	5	2	3
103	39	1882	1	4	1
104	40	1922	1	2	4
105	41	1963	6	1	4
106	42	2005	2	3	3
107	42	2047	5	1	3
108	43	2090	6	2	1
109	44	2134	3	1	2
110	45	2179	4	5	2
111	45	2224	7	5	3
112	46	2270	7	4	1
113	46	2316	6	4	4
114	47	2363	3	2	1
115	48	2411	4	3	3
116	49	2460	3	4	4
117	50	2510	1	5	3
118	50	2560	5	4	4
119	51	2611	3	1	4
120	52	2663	7	4	2
121	53	2716	4	4	1
122	53	2769	7	3	4
123	54	2823	3	5	1
124	55	2878	3	5	3
125	56	2934	2	3	4
126	57	2991	4	2	4
127	58	3049	7	5	2
128	59	3108	5	5	1
129	60	3168	3	1	1
130	60	3228	6	3	3
131	….	….	….	….	….
132	….	….	….	….	….
133	….	….	….	….	….
134	….	….	….	….	….

Let us consider $16 \in$ SN. It represents the ordered triple (R(16),C(16),P(16))=(6,3,1).Then D(16)= 7 and CD(16)=59.

7.3. VALUE OF THE WORD:

Let the value of the (partial) word defined as L_k .$V(L_k)$ is defined recursively.as$V(L_k)=V(L_{k-1})+V(a_i)$ with $V(L_0)=0$, where $D(a_i)$ is the distance array arranged such that $D(a_i) \leq D(a_{i+1})$. V (L_k) and V(X) the values of the pattern X will be the same. Since X is the (partial) pattern represented by L_4= (Sundara Murthy-1979).
Consider the partial word $L_4 =$ (1, 2, 3, 4)

$$= V (L_4) = 1+1+1+2= 5$$

7.4. LOWER BOUND OF A PARTIAL WORD LB (L_k)

A lower bound LB (L_k) for the values of the block of words represented by $L_k = \{a_1, a_2, - - - - , a_k\}$ can be defined as follows.

$$LB (L_k) = V (L_k) + \sum_{i=1}^{2n0-k} D(a_{k+j})= V (L_k) + CD (a_k + n_0 - k) - CD (a_k)$$

$$LB (L_4) = V (L_4) + CD (a_4 + n_0 - k) - CD (a_4)$$

$$= \mathbf{5+} CD (4+10 - 4) - CD (4)$$

$$=5+ CD (10) - CD (4)$$

$$= 5+ 24 – 5= 24$$

Where $CD (a_k) = \sum_{i=1}^{k} D(a_i)$. It can be seen that LB (L_k) is the value of partial word, which is obtained by concatenating the first (n-k) letters of SN (a_k) to the partial word L_k .i.e.L_4= {1,2,3,4 }

7.5. FESIABILITY CRITERIA OF PARTIAL WORD:

An algorithm was developed, in order to check the feasibility of a partial word $L_{k+1} = (a_1, a_2, - - - -- a_k, a_{k+1})$ given that L_k is a feasible word. We will introduce some more notations which will be useful in the sequel.

IR (R (a_i)) = 1, i = 1, 2, - - - - - , n and IR (j) = 0 for other elements of j.

L (i) = 1 ,i = 1, 2, - - - - -, n and L (j) = 0 for other elements of j

The recursive algorithm for checking the feasibility of a partial word L_p is given as follows. In the algorithm first we equate IX = 0, at the end if IX = 1 then the partial word is feasible, otherwise it is infeasible. For this algorithm we have RA=R (a_i), CA=C (a_i).

8. ALGORITHMS

8.1. Algorithm 1: (Checking for the Feasibility)

102. IX = 0 go to 104

104. IS (IR (RA) $\geq$2) if yes go to126

if no go to 106

106. IS (MX (CA,KA)=0 if yes go to 108

if no go to126

108. IS (JX (RA, 1) = 0) if yes go to 110

if no go to116

110 .L=JP (RA,1) go to 112

112. IS L = 1 if yes PX=1 go to 124

if no go to114

114. IS L = 2 if yes PX=1 go to 124

if no go to116

116. IS JX (RA, 2) = 0 if yes go to 118

if no go to126

118.L=JP(RA,2) go to120

120. IS L = 2 if yes PX=2 go to 124

if no go to122

122. L = 3 PX=2 go to 124

124 .IX = 1

126.STOP.

8.2. ALGORITHM – II (Lexi-Search)

STEP 1: (Initialization)

The arrays SN, D, CD, R, C,P, JP, JX, MX, LP,IR are made available.

RA,CA,KA,V,LB and n_0 are initialized to zero.

I = 1 J = 0, NZ=0 VT=999,n=8, $n_{0=}5$,m=5,p=4 and Max=nxmxr

STEP 2: J = J + 1

IS (J $\geq$ Max) if yes go to 13

if no go to 3

STEP 3: L(I)=J

RA = R(J)

CA = C(J)

KA = K(J)

go to 4

STEP 4: V(I) = V(I - 1) + D(J)

LB = V(I) + CD (J + 2 n_o – I) – CD(J)

go to 5

STEP 5: IS (LB $\geq$ VT)

if yes go to 13

if no go to 6

STEP 6: Check the Feasibility using Algorithm -I

IS (IX = 1)

if yes go to 7

if no go to 2

STEP 7:

LP(I) = PX

go to 8

STEP 8: L(I) = J

IR(RA)=IR(RA)+1

MX(CA, KA) = 1

JX(RA, PX) = 1

go to 9

STEP 9: count = IR(i > 0) (i=1,2….n)

go to 10

STEP 10 : Is n_0 = count

if yes go to 12

if no go to 11

STEP 11 : I = I+1 go to 2

STEP 12 : VT = V(I) L(I) = j.
Record VT go to 13

STEP 13: I = I-1 go to 14

STEP 14 : J = L(I)

IR(RA)=IR(RA)-1

MX(CA, KA) = 0

JX(RA, PX) = 0

LP(I) = 0

L(I+1) = 0 go to 2

STEP 15 : If I= 1

if yes go to 16

if no go to 13

STEP 16 : STOP

This recursive algorithm is used in Lexi search algorithm to check the feasibility of a partial word. We start the algorithm with a big value say '∞' as a testing value VT. If the value of a feasible word is known, we can as well start with that value as VT. During the search the value of VT is improved. At the end of the search the current value of VT gives the optimal feasible solution. We start the partial word $L_1 = (a_1) = (1)$. A partial word L_k is constructed as $L_k = L_k\text{-}1 * (a_k)$ where * indicates concatenation i.e. chain formation. We will calculate the values of V (L_k) and LB (L_k) simultaneously. Then two cases arise one for branching and the other for continuing the search.

1. If **LB (L_k) < VT**. Then we check whether L_k is feasible or not. If it is feasible we proceed to consider a partial word of order (k+1), which represents a sub block of the block of words represented by L_k. If L_k is not feasible then consider the next partial word of order by taking another letter which succeeds a_k in the k^{th} position. If all the words of order 'k' are exhausted then we consider the next partial word of order (k–1).

2. If **LB (L_k) ≥ VT.** In this case we reject the partial word L_k. We reject the block of word with L_k as leader as not having optimum feasible solution and also reject all partial words of order 'k' that succeeds L_k.

Now we are in a position to develop a Lexi-Search algorithm to find an optimal feasible word.

The current value of VT at the end of the search is the value of the optimal word. At the end if VT= ∞, it indicates that there is no feasible assignment.

9. SEARCH TABLE:

The working details of getting an optimal word, by using the above algorithm for the illustrative numerical example are given in the table-13. The column s R, C, P and remarks are acceptability of the partial words. In the following table-13, A indicates ACCEPT and R indicates REJECT.

Table-12 (Search table)

S.NO.	1	2	3	4	5	6	7	8	9	10	V	LB	R	C	P	REM
1	1										1	24	2	4	2,2	A
2		2									2	24	6	1	3,3	A
3			3								3	24	8	2	4,2	A
4				4							5	24	5	3	2,2	A
5					5						7	24	4	5	4,2	A
6						6					10	24	4	2	1,3	A
7							7				13	24	3*	4	3,1	R
8							8				13	26	3*	3	4,2	R
9							9				14	28	2	1	1,3	A
10								10			18	28	7*	3	2,3	R
11								11			19	29	8	5	1,1	A
12									12		24	3	6	2	2,2	A
13										13	30	30	5	5	3,1	A=VT
14									13		25	31*	5	5	3,1	R,>VT
15								12			19	31*	6	2	2,2	R,>VT
16							10				14	30*	7*	3	2,3	R,=VT
17						7					10	26	3*	4	3,1	R
18						8					10	28	3*	3	4,2	R
19						9					11	31*	2	1	1,3	R,>VT
20					6						8	27	4	2	1,3	A
21						7					11	27	3*	4	3,1	R
22						8					11	29	3*	3	4,2	R
23						9					12	32*	2	1	1,3	R,>VT
24					7						8	29	3	4	3,1	A
25						8					11	29	3	3	4,2	A
26							9				15	29	2	1	1,3	A
27								10			19	29	7*	3	2,3	R
28								11			20	31*	8	5	1,1	R,>VT
29							10				15	31*	7*	3	2,3	R,>VT
30						9					12	32*	2	1	1,3	R,>VT
31					8						8	32*	3	3	4,2	R,>VT
32				5							5	27	4	5	4,2	A
33					6						8	27	4	2	1,3	A

34						**7**					**11**	**27**	**3**	**4**	**3,1**	**A**
35							**8**				**14**	**27**	**3**	**3**	**4,2**	**A**
36								**9**			**18**	**27**	**2**	**1**	**1,3**	**A**
37									**10**		**22**	**27**	**7***	**3**	**2,3**	**R**
38									**11**		**23**	**28**	**8**	**5**	**1,1**	**A**
39										**12**	**28**	**28**	**6**	**2**	**2,2**	**A=VT_2=28**
40									**12**		**23**	**29***	**6**	**2**	**2,2**	**R,>VT**
41								**10**			**18**	**28***	**7***	**3**	**2,3**	**R,=VT**
42							**9**				**15**	**29***	**2**	**1**	**1,3**	**R,>VT**
43						**8**					**11**	**29***	**3**	**3**	**4,2**	**R,>VT**
44					**7**						**8**	**29***	**3**	**4**	**3,1**	**R,>VT**
45				**6**							**6**	**30***	**4**	**2**	**1,3**	**R,>VT**
46			**4**								**4**	**28***	**5**	**3**	**2,2**	**R,=VT**
47		**3**									**2**	**28***	**8**	**2**	**4,2**	**R,=VT**
48	**2**										**1**	**28***	**6**	**1**	**3,3**	**R,=VT**

10. COMENTS:

At the end of the search the current value of VT is 28 and it is the value of the feasible word and is given in 39th row of the search table – 12 and the corresponding order triples are

***(2,4,2), (6,1,3),(8, s2,4),(4,5,4),(4,2,1), (3,4,3), (3,3,4), (2,1,1) , (8,5,1)*,(6,2,2).**

The ordered triple set {***(2,42), (6,1,3), (8,2,4), (4,5,4), (4,2,1), (3,4,3), (3,3,4), (2,1,1), (8,5,1)***, **(6,2,2)}** represents the pattern given in the table-13-16, which is a optimal solution for the above numerical example.

Table - 13 ***Table - 14***

$$\mathbf{X(i,j,1)}=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \boldsymbol{X(i,j,2)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

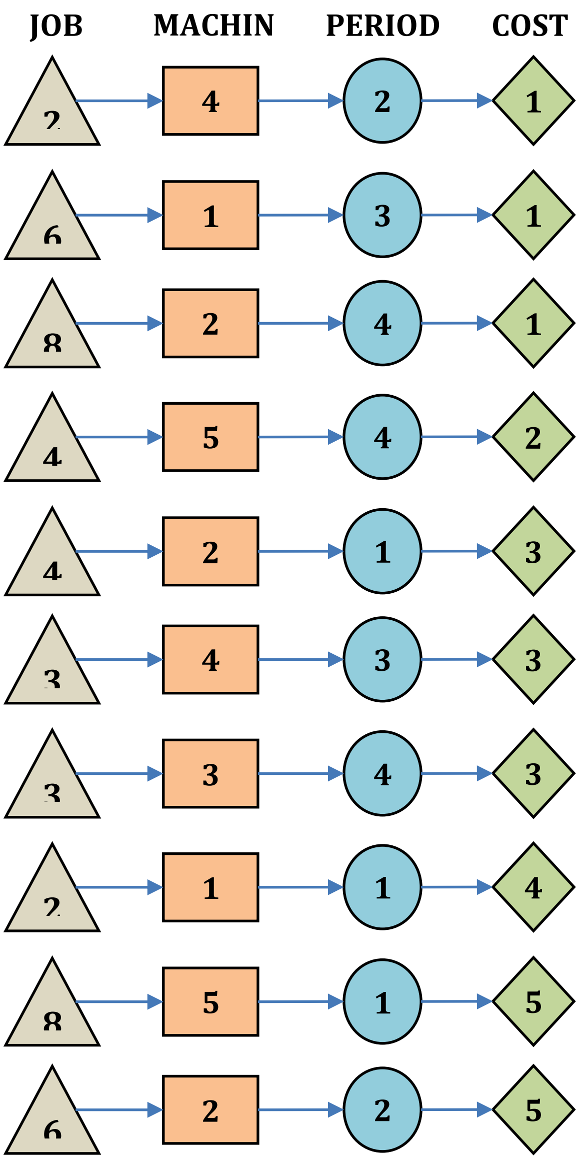
Table-15 ***Table-16***

X(i,j,3)=

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{X(i,j,4)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Figure-2 (Optimal Solution)

The following **figure-2** represents an optimal solution. The triangle shapes represent jobs, rectangle shapes represent machines, circle shapes represent periods and rhombus shapes represent costs. The values in triangle indicate names of the jobs, values in rectangles indicate names of the machines and value in the circle indicates number of period and value in rhombus indicates the corresponding cost of machine, job and period.



The above **figure-2** satisfies all the constraints in mathematical formulation . The ordered triples represents the assigning cost D(***2,42)=1, D(6,1,3)=1, D(8, 2,4)=1,D (4,5,4)=2, D(4,2,1)=3, D(3,4,3)=3, D(3,3,4)=3, D(2,1,1=4, D(8,5,1)=5***, *D*(**6,2,2)=5.**

Therefore the total cost= 01+01+01+02+03+03+03+04+05+05 = 28

11. CONCLUSION:

In this paper, we have studied a model namely **Multi Process Three Dimensional Job Assignment Model**. We developed a new algorithm which is efficient, accurate and easy to understand. First the model is formulated in to a zero- one programming problem. In this assignment problem it deals with a set of jobs with required processes done on a set of machines and periods. So it is a three dimensional multi process assignment problem. Though it is a multi process assignment problem we still treated it as a zero-one programming problem and develop Lexi-Search algorithm using pattern recognition technique for getting optimal solution. Now we are confident that we can solve higher dimensional problems also by this method.

REFERENCES

[1] *Balakrishna.U(2009)-GTSP,GTDSP & ASP Models,Ph.D Thesis,S.V.University, Tirupati*

[2] *Barr, R.S., Glover, F. & Klingman, D, the alternating basis algorithm for assignment Problem, Math. Prog., 13. 1-13, 1977.*

[3] *Bertsekas, D.P., A New Algorithm for the Assignment problem, Math. Prog.21,152-171, 1981.*

[4] *Garfinkel,R.S(1971):"An improved algorithm for bottleneck assignment problem",Operations Research 18,pp-(1717-1751)*

[5] *Geetha, S. & Nair, K.P.,A variation of the Assignment problem, Euro.J. Of Ors, 68, 422-426, 1993.*

[6] *Hung, M.S.& Rom,W.O., Solving the Assignment problem by relaxation. Ops.Res., 18, 969-982, 1980.*

[7] *Kuhn, H.W, the Hungarian Method for the Assignment Problem, Navl Rec. Log.Qtly.2, 83-97, 1955*

[8] *Purusotham, S et.al. International Journal of Engineering Science and Technology (IJEST Vol. 3 No. 8, 2011*

[9] *Ross G.T. and Soland, R.M1975):"A Branch and Bound algorithm for Generalized Assignment Problem-mathematical program-ming", pp:91-103.*

[10] *Sobhan Babu. K et.al. International Journal on Computer Science and Engineering, (IJCSE), Vol. 02, No. 05, 2010, 1633-1640.*

[11] *Subramanyam,Y.V.(1979):"Some special cases of assignment problem", Op search, vol:16(1),pp:45-47.*

[12] *Sundara Murthy, M - (1979). Combinatorial Programming - A Pattern Recognition Approach. PhD, Thesis REC, Warangal, India.*

[13] *Suresh Babu C, Purusotham, S. and Sundara Murthy, M (2008), Pattern Recognition based Lexi-Search Approach to the Variant Multi-Dimensional Assignment Problem" in the International Journal of Engineering Science and Technology. Vol. 3(8), 2011.*

[14] *Vidhyullatha, A Thesis(Pattern Recognition Lexi-Search Exact Algorithms For Variant ASP and Bulk TP models), Department of Mathematics, Sri Venkateswara University, Tirupati, 2011.*