Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 4; April-June, 2015 pp. 398-401

http://www.krishisanskriti.org/acsit.html

A Multi-Objective Distributed Job Shop Scheduling Problem: Solution by Multi-Objective Optimization Algorithm

Susmita Bandyopadhyay[1] and Arindam Kumar Chanda[2]

[1]*The University of Burdwan Burdwan-713104 West Bengal, India*
[2]*G.B. Pant Govt. Engineering College Okhla Phsae III Road New Delhi-110020, India*
E-mail: [1]bandyopadhyaysusmita2010@gmail.com, [2]akcped@yahoo.com

Abstract—*The age of globalization has changed the traditional single-facility manufacturing to multi-facility manufacturing system. This paper proposes a novel multi-objective multi-facility distributed job shop scheduling problem. The objectives are – 1) minimization of makespan and 2) minimization of tardiness of jobs. The proposed multi-objective problem is formulated with the aim of optimizing a set of time-related objectives in such scenario. The proposed problem has been solved by a nature-based technique and efforts have been exerted to make better use of the proposed nature based technique. A new mutation algorithm based on the mechanism of circular queue has been proposed and embedded in the proposed nature based algorithm. Experimental results show the effectiveness of the proposed solution technique for the proposed multi-objective problem.*

Keywords: *Distributed Job Shop Scheduling; Multi-objective nature-based algorithm; Mutation Algorithm; Triangular Fuzzy number; Makespan; Tardiness*

1. INTRODUCTION

The traditional production system is a single factory production system where all jobs are processed. With the increase of complexity in the manufacturing problems, the single factory system has been evolved into decentralized multi-factory system. Such distributed multi-factory system has several advantages including reduction in production cost, enhancement of production capability, proximity to the target market and so on. The current research study is a preliminary effort towards this direction. In the proposed distributed job shop situation, a set of identical factories is proposed. The jobs in a job order are distributed into the factories. Such problems are difficult to handle since two main decisions have to be taken – 1) allocation of jobs to factories and 2) scheduling jobs in each factory. The research problem as proposed in this paper deals with this type of problem where a bi-objective problem is proposed. The objectives are – 1) minimizing makespan and 2) minimizing tardiness of jobs. The processing time has been assumed to be triangular fuzzy numbers. The makespan here is the completion time of all jobs in a job order whose jobs have been distributed among the factories. The proposed bi-objective problem has been solved by a proposed nature-based algorithm. A circular mutation algorithm has been proposed and embedded into the proposed nature-based algorithm.

2. LITERATURE REVIEW

The existing literature shows some research studies on distributed job shop scheduling problems, although the number of research studies on multi-objective and multi-facility job shop scheduling problem is really very few. Chan et al. [1, 2] also applied Genetic Algorithm for solving distributed flexible manufacturing problem. Jia et al. [3, 4] also proposed a distributed job shop problem and solved the proposed problem by Genetic Algorithm. Naderi and Ruiz [5] developed 6 different mathematical models on distributed flow shop scheduling problem. Behnamian and Fatemi Ghom [6] applied Genetic Algorithm to solve distributed parallel machine problem. Some of the other significant research studies include research studies of Gao [7], Liu and Gao [8], Gao et al. [9], Gao and Chen [10], Lin etal. [11], Wang et al. [12].

3. PROBLEM FORMULATION

Before presenting the formulation of the problem and the related explanations, the assumptions and notations are presented.

3.1 Assumptions

The basic assumptions are enlisted below.

1. Each job visits all the machines in a factory
2. Jobs are randomly allotted to the factories
3. The starting time at which a job enters a factory is assumed to be zero which can represent any time of a day.
4. Processing time of a job processed in a machine is a triangular fuzzy number.
5. Due dates of the jobs are fixed.

3.2 Notations

Decision Variables

Z_{if} : 1, if job *i* is allotted to factory *f*, *0* otherwise

y_{imf} : 1, if job *i* is assigned to machine *m* in factory *f*

x_{ijmf} : 1, if job *j* follows job *i* on machine *m* in factory *f*

Parameters

J : Total number of jobs in a job order

F : Total number of factories

J_f : Total number of jobs allotted to factory *f*

M_f : Total number of machines in factory *f*

i, j : Subscript for jobs

m : Subscript for machines

c_{fi} : Completion time of job *i* in factory *f*

S_{jmf} : Starting time of job *j* on machine *m* in factory *f*

p_{jmf} : Processing time of job *j* on machine *m* in factory *f*

T_{if} : Tardiness of job *i* in factory *f*

d_{if} : Due date for job *i* in factory *f*

3.3 Formulation of the Problem

$$Minimize\ Z_1 = \max(c_{11}, c_{12}, ..., c_{1J_1}), \max(c_{21}, c_{22}, ..., c_{2J_2}), ..., \max(c_{F1}, c_{F2}, ..., c_{FJ_F}) \quad (1)$$

$$Minimize\ Z_2 = \sum_{f=1}^{F} \sum_{i=1}^{J_f} Z_{if} T_{if} \quad (2)$$

Subject to the constraints:

$$\sum_{i=1, i \neq j}^{J_f} Z_{if} Z_{jf} y_{imf} y_{jmf} x_{ijmf} = 1, \forall f \quad (3)$$

$$\sum_{j=1}^{J_f} Z_{if} Z_{jf} x_{ijmf} \leq y_{imf}, \forall f, \forall i \quad (4)$$

$$\sum_{m=1}^{M_f} y_{imf} = 1, \forall f, \forall i \quad (5)$$

$$c_{fi} \geq S_{jmf} + p_{jmf}, \forall f \quad (6)$$

$$x_{ijmf} + x_{jimf} = 1 \quad (7)$$

$$T_{if} \geq c_{fi} - d_{if} \quad (8)$$

Explanation

Objective (1) minimizes the makespan of the entire job order *J*, consisting of several jobs which have been distributed among *F* factories. Objective (2) minimizes total tardiness. Constraint (3) ensures that only one job precedes each job. Constraint (4) conveys that if job *j* follows job *i* then both job *i* and job *j* belong to machine *m*, assuming that only one job follows a job and only one job precedes a job. Constraint (5) indicates that each job is assigned to exactly one machine. Constraint (6) means that the completion of job *i* is greater than or equal to the starting time of the job on machine m plus the processing time of the same job on the same machine. Constraint (7) ensures that either job *i* will follow job *j* or job *j* will follow job *i*. constraint (8) defines tardiness of job *i*.

4. MODEL DESCRIPTION

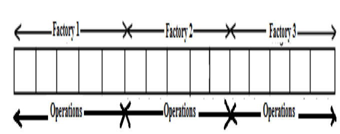
The formulated problem has been solved by a nature based algorithm where the only genetic operation is mutation. The purpose is to generate a large variety of sequences on which the makespan and tardiness depends. Thus, unlike the other genetics-based algorithms, this algorithm has only applied mutation. The overall algorithm is shown in Fig. 1. This algorithm is quite different from the general view of genetics-based algorithms since the entire population represents a job order. Thus each run of the same developed model represents a separate job order.

```
For each job order
    Assign J number of jobs to f number of factories. The size of the population is J.
    Evaluate objectives for each chromosome in the population
    Sort the population of chromosomes
    Apply mutation on each chromosome in the population and evaluate chromosome
End For
```

Fig. 1: Proposed Multi-Objective Nature-Based Algorithm.

First the population is generated using algorithm shown in Fig. 2. The genotype of the chromosomes is shown in Fig. 3. The sorting algorithm applied is the nondomination sorting. The mutation operation in this algorithm is responsible for increasing the variety of sequences in the population. The mutation algorithm applied is shown in Fig. 4. The Mutation algorithm just works like a circular queue as shown in Fig. 4. Fig. 5 clearly shows Mutation algorithm through an example. The original sequence in this example is 4 – 3 – 5 – 1 – 2. The Circular Mutation applies a circular motion to the left direction. As a result, each cell value shifts to left by 1 cell and since the motion is circular, thus the value of cell 1 shifts to the last cell. The algorithm continues till the last job in the entire job order.

```
Initialize processing time and duedates
For each job
    Assign the job a factory
    Generate the sequence of operations
End For
```



Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 4; April-June, 2015

Fig. 2: Algorithm for Initialization. Fig. 3: Genotype of Chromosome.

```
For each individual
    For each stage
        Store the value of cell 1 in 'store', that is, store ←Value[cell = 1]
        For cell ← 1 to 5
            If cell ~ = 5 Then
                Assign Value[cell] ← Value[cell+1]
            Else
                Assign Value[cell] ← store
            EndIf
        EndFor
    EndFor
EndFor
```

Fig. 4: Mutation Algorithm.

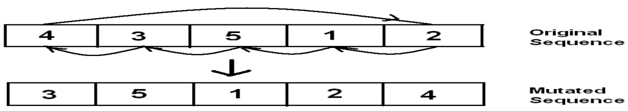


Fig. 5: Explanation of Mutation Algorithm.

5. RESULTS AND DISCUSSION

The experimentation has been done in a 2.8 GHz PC with 4 GB RAM. The software used is Matlab. For experimentation, a total of 3 factories are assumed. The maximum number of machine per factory is assumed to be 6. The job order is assumed to contain 50 jobs. Since the formulated problem is a multi-objective one, thus the first result to show is the Pareto optimal solutions which are being shown in Fig. 6. The model developed in Matlab has been run for 100 times and the results of the best 20 runs have been noted in particular.

In Fig. 6, the horizontal and vertical axes represent the tardiness of jobs and makespan respectively. The Fig. shows that the makespan is the lowest for factory 3 as the range of values of makespan for this factory is 0 – 25 whereas the range of values of makespan for factory 1 and factory 2 are 0 – 40 and 0 – 60 respectively. The makespan here represents the aggregate of the completion times of all the jobs in a particular job order, whose jobs are distributed among the three factories. The completion time of all the jobs assigned to a particular factory at a particular time as well as the makespan of all the jobs in a job order are both measured. A sample of best sequence of operations for three individual factories for three best completion times and best tardiness is shown in Fig. 7.

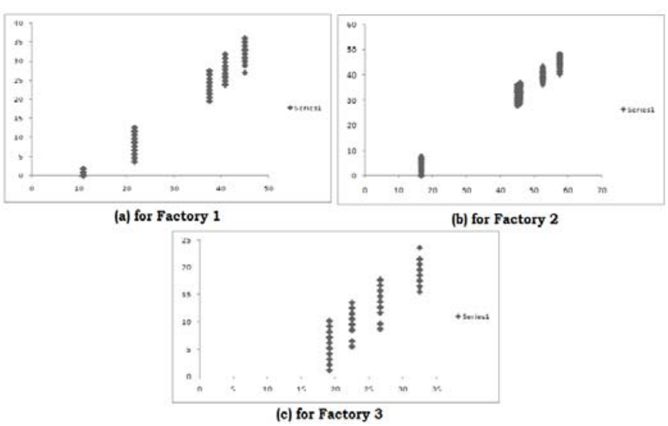


Fig. 6: Pareto Optimal Solutions for Factory 1, Factory 2 and Factory 3.

5	2	3	1	4
5	4	3	1	2
5	2	3	1	4
5	3	1	2	1
5	3	1	4	2
5	4	2	3	1
5	4	1	2	3
5	2	3	1	4
5	4	3	2	1
5	2	1	4	3
5	3	1	2	4
5	1	2	3	4
5	4	3	1	2
5	3	2	4	1
5	1	2	4	3
5	4	2	1	3
5	2	4	1	3
5	2	1	3	4
5	1	3	2	4
5	4	1	3	2
5	2	1	4	3

for factory 1

5	1	4	2	3
5	2	4	1	3

for factory 2

2	4	1	5	3
3	1	4	2	5
2	1	1	4	3

for factory 3

Fig. 7: A Sample of Best Sequence of Operations Representing Best Completion Time and Best Tardiness.

Besides the minimum and average values of makespan for the three factories for the 20 different job orders representing 20 runs is shown in Fig. 8. A sample Pareto Optimal solutions is shown in Fig. 9.

Job Order	Minimum Makespan	Average Makespan
1	1519	1642
2	1537	1624
3	1487	1661
4	1631	1681
5	1525	1742
6	1597	1643
7	1537	1652
8	1553	1583
9	1393	1517
10	1436	1529
11	1519	1570
12	1491	1624
13	1473	1534
14	1498	1600
15	1481	1550
16	1537	1630
17	1492	1567
18	1484	1611
19	1288	1599
20	1502	1567

Fig. 8: Minimum and Average Values of Makespan for 20 Different Job Orders

21.667	7.6667
26.667	11.667
45.833	29.833
40.833	23.833
37.5	19.5
45	32
45	31
10.833	0
19.167	3.1667
19.167	2.1667
10.833	0
26.667	14.667
10.833	0
57.5	42.5
26.667	12.667
16.667	3.6667
21.667	5.6667
21.667	7.6667
19.167	8.1667
21.667	8.6667

Fig. 9: Sample Pareto Optimal Solutions

6. CONCLUSION

This paper has proposed a distributed multi-objective job shop scheduling having two objectives – minimizing makespan and minimizing tardiness. The distributed manufacturing problem is defined by the distribution of the jobs in a job order into a set of identical factories. The processing times are assumed to be triangular fuzzy numbers. The proposed bi-objective problem has been solved by a multi-objective nature-based algorithm. The population in the proposed multi-objective algorithm represents a job order whose jobs are distributed among the factories. A new mutation algorithm has been proposed and embedded in the proposed multi-objective nature-based algorithm. The mutation algorithm works like the functioning of a circular chromosome where the genes in chromosome shift their position in a circular fashion.

Although the research study in the distrusted job shop problem is not new in the existing literature, but the multi-objective treatment of such problem is not at the significant level, as evident from the existing literature. the research study presented in this paper is an effort towards this direction with an aim of significant improvement of the algorithmic applications in this area of research. Thus the research study as proposed in this paper can be extended can be extended in order to make it closer to reality.

REFERENCES

[1] Jia, H.Z., Fuh, J.Y.H., Nee, A.Y.C. and Zhang, Y.F., "Web-based multi-functional scheduling system for a distributed manufacturing environment", *Concurrent Engineering—Research and Applications*, 10, 1, 2002, pp. 27–39.

[2] Jia, H.Z., Nee, A.Y.C., Fuh, J.Y.H. and Zhang, Y.F., "A modified genetic algorithm for distributed scheduling problems", *Journal of Intelligent Manufacturing*, 14, 3-4, 2003, pp. 351–362.

[3] Jia, H.Z., Fuh, J.Y.H., Nee, A.Y.C. and Zhang, Y.F., "Integration of genetic algorithm and Gantt chart for job shop scheduling in distributed manufacturing systems", *Computers and Industrial Engineering*, 53, 2, 2007, pp. 313–320.

[4] Chan, F.T.S., Chung, S.H. and Chan, P.L.Y., "An adaptive genetic algorithm with dominated genes for distributed scheduling problems", *Expert Systems with Applications*, 29, 2, pp. 364–371.

[5] Behnamian, J. and Fatemi Ghom, S.M.T., "The heterogeneous multi-factory production network scheduling with adaptive communication policy and parallel machine", *Information Sciences*, 219, 2013, pp. 181–196.

[6] Naderi, B. and Rubén, R., "The distributed permutation flowshop scheduling problem", *Computers and Operations Research*, 37, 2010, pp. 754-768.

[7] Gao, J., Chen, R., Deng W. and Liu Y., "Solving multi-factory flowshop problems with a novel variable neighbourhood descent algorithm", *Journal of Computational Information Systems*, 8, 5, 2012, pp. 2025-2032.

[8] Liu, H. and Gao, L., "A discrete electromagnetism-like mechanism algorithm for solving distributed permutation flowshop scheduling problem", *IEEE International Conference on Manufacturing Automation 2010*, DOI 10.1109/ICMA.2010.17.

[9] Gao, J., Chen, R. and Liu, Y., "A knowledge-based genetic algorithm for permutation flowshop scheduling problems with multiple factories", *International Journal of Advancements in Computing Technology*, 4, 7, 2012, pp. 121-129.

[10] Gao, J. and Chen, R., "A hybrid genetic algorithm for the distributed permutation flowshop scheduling problem", *International Journal of Computational Intelligence Systems*, 4, 4, 2011, pp. 497-508.

[11] Lin, S.W., Ying, K.C. and Huang, C.Y., "Minimising makespan in distributed permutation flowshops using a modified iterated greedy algorithm", *International Journal of Production Research*, 51, 16, 2013, pp. 5029-5038.

[12] Wang, S., Wang, L., Liu, M. and Xu, Y., "An effective estimation of distribution algorithm for solving the distributed permutation flow-shop scheduling problem", *International Journal of Production Economics*, 145, 1, 2013, pp. 387-396.