Applied Mechanics and Materials
ISSN: 1662-7482, Vol. 591, pp 184-188
doi:10.4028/www.scientific.net/AMM.591.184
© 2014 Trans Tech Publications, Switzerland

Submitted: 2014-05-12
Revised: 2014-06-02
Accepted: 2014-06-02
Online: 2014-07-18

A New GT Heuristic for Solving Multi Objective Job Shop Scheduling Problems

D.Lakshmipathy[1*], M.Chandrasekaran[2,b], T.Balamurugan[3,c], P.Sriramya[4,d]

[1]Research Scholar, Dept of Mechanical Engineering, Vels University, Chennai, India
[2] Professor, Dept of Mechanical Engineering, Vels University, Chennai, India
[3] Asso. Prof, St.Joesph's College of Engineering, Chennai, India
[4] Asso. Prof, Saveetha School of Engineering, Chennai, India
lakshmipathid@yahoo.co.in (Corresponding author)

Keywords: Job Shop Scheduling, Multi Objectives, GT Heuristic.

ABSTRACT

The n-job, m-machine Job shop scheduling (JSP) problem is one of the general production scheduling problems in manufacturing system. Scheduling problems vary widely according to specific production tasks but most are NP-hard problems. Scheduling problems are usually solved using heuristics to get optimal or near optimal solutions because problems found in practical applications cannot be solved to optimality using reasonable resources in many cases. In this paper, optimization of three practical performance measures mean job flow time, mean job tardiness and makespan are considered. New Game theory based heuristic method (GT) is used for finding optimal makespan, mean flow time, mean tardiness values of different size problems. The results show that the GT Heuristic is an efficient and effective method that gives better results than Genetic Algorithm (GA). The proposed GT Heuristic is a good problem-solving technique for job shop scheduling problem with multi criteria.

1 INTRODUCTION

The classical job-shop scheduling problem (JSP) is one of most difficult combinatorial optimization problems. During the last decades a great deal of attention has been paid to solving these problems with many algorithms by considering single objective. But real world scheduling problems naturally involve multiple objectives. There are only few attempts available to tackle the multi-objective JSP. In a multi-objective context, find as much different schedules as possible, which are non-dominated with regard to two or more objectives. Some frequently used performance measures are makespan, mean flow-time and mean tardiness. Makespan is defined as the maximum completion time of all jobs. Mean flow-time is the average of the flow-times of all jobs. Mean tardiness is defined as the average of tardiness of all jobs.

2 LITERATURE REVIEW

2.1 Job Shop Scheduling

Bruker [1] show that the Job shop Scheduling (JSP) is an NP-hard [2] combinatorial problem. Because of the NP-hard characteristics of job shop scheduling, it is usually very hard to find its optimal solution, and an optimal solution in the mathematical sense is not always necessary in practices [3]. Researchers turned to search its near-optimal solutions with all kind of heuristic algorithms [4]. Fortunately, the searched near optimal solutions usually meet requirements of practical problems very well.

In a single-objective context some of the recent approaches have shown quite promising results [5-6]. But real world scheduling problems naturally involve multiple objectives. There are only few attempts to tackle the multi-objective JSP [7].

2.2 Graph Theory based (GT) Heuristic

Trees are more important data structures, which come in many forms. Some times trees are static in the sense that their shape is determined before running of the algorithm, and they do not change shape while the algorithm runs. In other cases, trees are dynamic, meaning that they undergo shape changes during the running of the algorithm.

3.2.1 Depth First Search

- Declare two empty lists: Open and Closed. Add Start node to open list.
- While Open list is not empty, loop the following:
    - a. Remove the first node from Open List.
    - b. Check to see if the removed node is destination.
- If the removed node is destination, break out of the loop, add the node to closed list, and return the value of closed list.
- If the removed node is not destination, continue the loop (go to Step c).
- Extract the neighbors of above removed node.
- Add the neighbors to the *beginning* of Open list, and add the removed node to closed list. Continue looping.

3.2.2 Breadth First Search

- Declare two empty lists: Open and Closed. Add Start node to open list.
- While Open list is not empty, loop the following:
- Remove the first node from Open List.
- Check to see if the removed node is destination
- If the removed node is destination, break out of the loop, add the node to closed list, and return the value of closed list.
- If the removed node is not destination, continue the loop (go to Step c).
- Extract the neighbors of above removed node.
- Add the neighbors to the *end* of Open list, and add the removed node to closed list.

3.2.3 Tree traversal

In computer science, tree traversal is the process of visiting each node in a tree data structure. Tree traversal, also called walking the tree, provides for sequential processing of each node in what is, by nature, a non-sequential data structure. Such traversals are classified by the order in which the nodes are visited.

If n is a node in a binary search tree, then everything in n's left subtree is less than n, and everything in n's right subtree is greater than or equal to n. Thus, the left subtrees in order, using a recursive call, and then visit n, and then visit the right subtree in order. Assume the recursive calls correctly visit the subtrees in order using the mathematical principle of structural induction.

3 PROBLEM FORMULATION

In a multi-objective context, find as much different schedules as possible, which are non-dominated with regard to two or more objectives. Performance measures are makespan, mean flow-time and mean tardiness. Makespan is defined as the maximum completion time of all jobs. Mean flow-time is the average of the flow-times of all jobs. Mean tardiness is defined as the average of tardiness of all jobs.

The combined objective function for the multi objective Job Shop Problem is,
COF=Min [w1 (msi/ms*) + w2 (Ti/T*) + w3 (mfi/mf*)]
Where, w1 = (R1/∑R), w2 = (R2/∑R),
w3 = (R3/∑R) ∑R = (R1+ R2 +R3),
where R1, R2, R3 - Random numbers
ms*- Make Span Global minimum
T* - Mean Tardiness Global minimum
mf* - Mean Flow Time Global minimum
ms_i - Make span Iteration minimum
Ti - Mean Tardiness Iteration minimum
mfi - Mean Flow Time Iteration minimum
w1, w2, w3- Weightage factors
MFT- Mean flow time,
MT - Mean Tardiness
COF- Combined Objective Function

5 RESULTS AND DISCUSSION

The new heuristic is developed and implemented in C language on personal computer Pentium IV 2.4 GHz. The maximum number of iterations has been set to 100 X n, where n is the number of jobs. Multi-objective optimization differs from single-objective optimization in many ways [8]. For two or more conflicting objectives, each objective corresponds to a different optimal solution, but none of these trade-off solutions is optimal with respect to all objectives. Thus, multi-objective optimization does not try to find one optimal solution but all trade-off solutions.

For multi-objective scheduling the proposed new GT heuristic is used to optimize makespan, mean flow time and mean tardiness of the two JSP given by Bagchi [7] are the basis of the following experiments. The first problem, called JSP1, is a ten job five machine instance. The second problem, called JSP2, is a ten job ten machine instance. Apparently, the GT heuristic algorithm minimizes all objectives simultaneously. This GT heuristic is compared with the similar previous work using GA [9] and shown in Table 1 and Table 2.

Notations Used

MS - Makespan, MFT - Mean Flow Time
MT - Mean Tardiness, COF - Combined Objective Function

Table 1. Results of GT Heuristic with Genetic Algorithm for JSP 1

JSP 1	Genetic Algorithm				**GT Heuristic**			
	MS	MT	MFT	COF	**MS**	**MT**	**MFT**	**COF**
1	156	10.8	128.4	0.926	**152**	**16.845**	**117.8**	**0.4811**
2	158	8.2	126	0.903	**162**	**16.805**	**111.2**	**0.3450**
3	159	15.7	124.3	0.622	**163**	**15.802**	**128.5**	**0.5674**
4	159	7.8	127.3	0.773	**164**	**12.97**	**115.28**	**0.5714**
5	160	13.9	124.3	0.630	**168**	**12.042**	**114.7**	**0.3696**
6	162	6.4	130.5	0.672	**170**	**14.985**	**121.65**	**0.6167**
7	165	6.4	128.8	0.867	**172**	**10.442**	**113.42**	**0.3861**
8	167	15.1	122.4	0.629	**170**	**12.804**	**119.58**	**0.6112**
9	169	6.1	134.5	0.683	**171**	**7.822**	**119.91**	**0.6729**
10	182	5.8	135.4	0.632	**178**	**9.759**	**139.30**	**0.6023**

6 CONCLUSION

In this paper, a new heuristic approach has been used for solving multi objective job shop scheduling problems with the objective of minimization of makespan, mean flow time and mean tardiness. This approach uses simple but effective techniques depth first search, Breadth first search and tree traversal. This approach has been tested on JSP 1 and JSP 2 problem instances given in Bagchi [7]. The findings were compared with Genetic Algorithm [12] that tested the same problems. The New heuristic gives better results than the genetic algorithm. The proposed new heuristic is competent and proves to be a good problem-solving technique for job shop scheduling.

Table 2 Results of GT Heuristic with Genetic Algorithm for JSP 2

JSP 2	Genetic Algorithm				**GT Heuristic**			
	MS	MT	MFT	COF	**MS**	**MT**	**MFT**	**COF**
1	196	32.2	174.7	0.726	**185**	**30.605**	**152.10**	**0.6821**
2	199	33	174.6	0.703	**196**	**31.597**	**138.2**	**0.4278**
3	201	31.8	176.1	0.622	**205**	**31.910**	**107.3**	**0.4805**
4	203	32.2	173.4	0.673	**206**	**31.965**	**166.25**	**0.3135**
5	204	31.3	174.8	0.630	**207**	**32.152**	**139.7**	**0.5023**
6	212	31.6	174.5	0.602	**208**	**30.086**	**147.1**	**0.5950**
7	228	30.7	189.1	0.667	**210**	**31.418**	**135.3**	**0.4964**
8	230	29.3	179.4	0.629	**213**	**31.398**	**148.31**	**0.4030**
9	238	28.2	188.1	0.683	**220**	**32.413**	**137.89**	**0.3899**
10	254	29.2	186.7	0.632	**224**	**32.856**	**135.2**	**0.4578**

7 REFERENCES

[1] Bruker, P., Scheduling Algorithms 2nd Edn, Springer-Verlag, Berlin (1995).

[2] Garey, M., et al., The complexity of flow shop and job shop scheduling, Mathematics of Operations Research, 1, (1976) pp 117-129.

[3] Erschler,J.F., Roubellat,J.P., Vernhes., Finding some essential characteristics of the feasible solutions for a scheduling problem. Operations Research, 24, (1976) pp 774-783.

[4] French, S., Sequencing and scheduling: An introduction to the mathematics of the job shop, New York, Wiley (1982).

[5] Mattfeld, D.C., Evolutionary Search and the Job Shop, Physica-Verlag, (1996).

[6] Ono, I., Yamamura, M., and Kobayashi, S., A genetic algorithm for job-shop scheduling problems using job-based order crossover, In Proceedings of ICEC '96, (1996), pp 547-552.

[7] Bagchi, T.P., Multiobjective Scheduling By Genetic Algorithms, Kluwer Academic Publishers, (1999)

[8] Deb, K., Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, (2001).

[9] Garen, J., Multi objective Job-Shop Scheduling with Genetic Algorithms Using a New Representation and Standard Uniform Crossover, MH Workshop, (2003).