International Journal of Engineering Research and General Science Volume 3, Issue 3, May-June, 2015
ISSN 2091-2730

Optimization of Make-span and Total Tardiness for Flow shop Scheduling Using Genetic Algorithm

P. Muni Babu, B.V. Himasekhar sai, Dr. A. Sreenivasulu Reddy

Department of Mechanical Engineering, S.V University College of Engineering, Tirupati, A.P, India

E-mail:munibabu29@gmail.com

Abstract— The Flow Shop Scheduling refers to the schedule planning problems especially for larger volume of systems with very less variations in requirements. Make-span and total tardiness are two most important goals in scheduling to make the given schedule plan as an efficient and should be able to satisfy the customer demands. Flow shop scheduling become NP Hard problem due to its larger size and laborious operations; moreover the bigger problem need more time to solve. In this paper genetic algorithm is used to solve n-jobs m-machines flow shop scheduling problem to get the optimum results of make-span and total tardiness. A JAVA program is developed for this scheduling problem, where the key operation for obtaining the optimum results is coded by GA(Genetic Algorithm). The present work considers two case studies one of them is that all jobs are required for every machine in a shop floor, and the second case is a job may not require to vary machine. In each case four simulation runs are performed for each combination of crossover and mutation in order to optimize the make-span, total tardiness and therefore finally to find the required job sequence.

Keywords— Flowshop Scheduling, Genetic Algorithm, NP Hard, Make-span, Total Tardiness, Java.

INTRODUCTION

Scheduling is one of the critical issues in a manufacturing system. The problem in scheduling focuses on how to allocate the limited resources of production, such as machinery, material handling, operators, and other equipment to carry out the process in a series of operating activities (job) in a certain period of time to optimize certain objective function.

Flow shop scheduling problem is Non-Polynomial hard (NP-Hard) problem because the bigger problem requires more time to get the optimal solution. Thus, the use of exact methods such as branch and bound, linear programming and Lagrangian relaxation is not effective enough and needs other method which is able to give effective in terms of results and computation time. The Flow Shop Scheduling involves where a set of 'n' jobs have to be processed with identical flow patterns on 'm' machines. The jobs have different processing time for different machines and jobs have different due date. Each job has to be processed in different stages in particular order. In this case arrangement of the jobs in a particular order is done to get many combinations and choose that combination where the minimum make-span and minimum total tardiness are achieved.

The flow shop scheduling problems become more complex with multi-objective. Therefore, it is required simultaneous consideration of multiple goals to generate the schedule so that it is able to optimize some objectives. Utility function approach is a method often used in multi objective problem, in which each objective will be given the weight suits in order of priority. The purpose of this work is to optimize two objective functions, make-span and total tardiness in flow shop scheduling.
Flow shop scheduling with multi objective involves several parameters, thus it leads to a combinational flow shop problem. In the present work, an attempt is made to optimize make span and total tardiness of Flow shop scheduling problem using Genetic Algorithm (GA) approach.

ASSUMPTIONS OF FLOW SHOP SCHEDULING

The flow shop scheduling problem considers the following assumptions:

- The operation processing times on the machines are known, fixed and some of them may be zero if some job is not processed on a machine.
- Set-up times for the operations are included in the processing times.
- Every job has to be processed on all machines in the order (j = 1, 2, ..., m).
- Every machine processes only one job at a time.
- Every job is processed on one machine at a time.
- Operations are not pre-emptive.

PROBLEM DESCRIPTION

In flow shop there are 'm' machines and 'n' jobs. Each job consists of m operations and each operation requires a different machine n-jobs have to be processed in the same sequence on m-machines. Processing time of job i on machine j is given by t_{ij} (i =1… n ; j =1,…,m), and each job due date D_i (i=1, 2….n).

A performance measures for scheduling is make-span (C_{max}) which has been used for maximum utilization of resources to increase productivity and stated as maximum completion time of last job to exit from the system.
C_{max} = Max (C_1 ,……., C_n).

Second performance measures for scheduling is Total tardiness is given as:

Total tardiness =$\sum T[i]$

Where $T[i] = C[i] – D[i]$ $\quad C[i] – D[i] \geq 0$

$= 0$ $\quad$ otherwise.

This has been used for satisfies customer demands.

The multi objective is to optimize both the make-span and total tardiness together. The main objective is achieved by introducing weightages for individual objective. The values of the weightages (w) can be fixed depending on priority. The values usually lie between 0 and 1. Equal priority for Make-span and Total Tardiness are considered in the present study.

Therefore multi-objective fitness function is obtained by combining the above two objectives into single scalar function so as to minimize make-span, total tardiness, which has been framed as:

$f(x)=Min [W_1C_{max}+W_2\sum T_{[i]}]$

Subjected to the constraints: $W_1 \geq 0$, $W_2 \geq 0$ And $W_1+W_2 = 1$

Where,

C_{max} = Makespan,

$\sum T_{[i]}$ = Total Tardiness,

W_1= Weight for Makespan,

W_2 = Weight for Total Tardiness.

ADOPTED METHODOLOGY

GA is inspired by Darwin's Theory about evolution - "survival of the fittest". GA represents an intelligent exploitation of a random search used to solve optimization problems. In the Simple GA-based approach, the various stages like evaluation, selection, crossover and mutation are repeatedly executed after initialization until a stopping criterion is met. The algorithm works on multiple solutions simultaneously. In this a general purpose schedule optimizer for manufacturing flow shops scheduling using genetic algorithms. Genetic Algorithm Procedure as follows

Step-1: Generate Initial Population
Initial solutions are randomly generated and these initial solutions form the first population.

Step-2: Record Optimal Solutions
Calculate the objective values of chromosomes in the population and record the optimal solutions.

Step-3: Calculate Objective Value
The total objective function is constituted of the linear combination of objective functions. And the weights are fixed depending on priority. For a solution x, the objective function in the study is represented as follows:

$$f(x)=[W_1C_{max}+W_2\sum T_{[i]}]$$

Where,

C_{max} = make-span

$\sum T_{[i]}$ = total tardiness

Step-4: Evaluate Fitness Value
For a solution x, its fitness equals to the minimum objective value in the generation itself.

Fitness value= $Min(W_1C_{max}+W_2 \sum T_{[i]})$

Step-5: Reproduction / Selection
The individuals from previous population are taken in which crossover and mutation has taken place.

Step-6: Crossover
In this study, two point crossover is used. For a crossover, two strings are to be selected randomly to make a pair for crossover.
For Example

String 1: 12 **345** 678
String 2: 23 **156** 487

Let crossover positions selected are after 2 and after 5. The elements between these are exchanged in parent strings, keeping other bits unchanged. So off springs produced are

New String 1 after crossover: 12 **156** 678
New String 2 after crossover: 23 **345** 487

Step-7: Mutation
In the present work, Position Based mutation is used. .
Example:
String before mutation: 13**2**46875

Let the randomly selected job is 2 and position after mutation is 5. Thus the string after mutation is
So string after mutation: 1346**2**875

Step-8: Replacement
The new population generated by the previous steps updates the old population.

Step-9: Update Optimal Solutions
Search the optimal solutions in the new population and update the old optimal solutions with new ones.

Step-10: Stopping Rule
If the number of generations equals to the pre-specified number then stop, otherwise go to step-5.

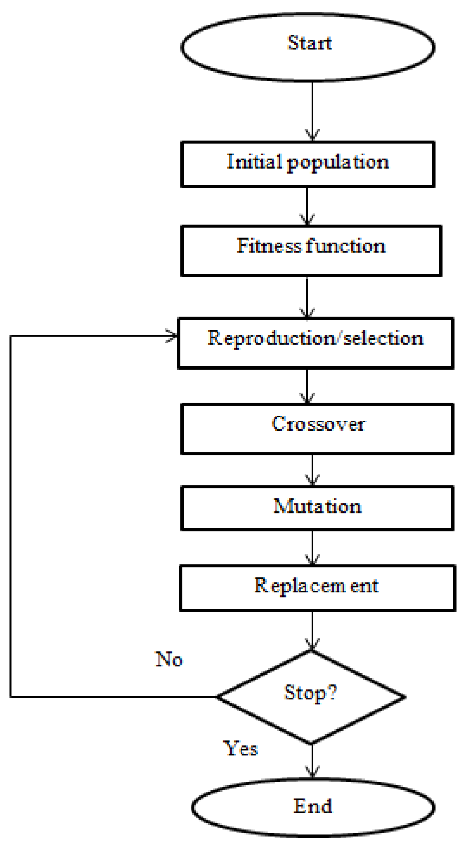

Figure.1 Flow Chart for Genetic Algorithm

RESULTS AND DISCUSSION

In this section two examples are presented to evaluate the proposed method. To perform experimental the presented algorithm is coded in java and executed.

For case studies of flow shop, optimization is carried out by taking into consideration, four combinations of iterations 50, 75, 100, 150. Simulations runs for each combination of parameter are carried out and optimization yields the best make-span and total tardiness as well as jobs sequence among four simulation runs.

Case study-1

It is an 8 jobs and 5 machines flow shop problem has been taken into consideration Table 1 provides the details of processing times as well as each job due date information for the case study.

Table 1. Processing Time of Jobs on Machine and Job Due Date (All values are in minutes**)**

Job No.	M_1	M_2	M_3	M_4	M_5	Due Date
1	32	21	10	51	33	678
2	1	27	42	19	45	396
3	61	87	66	23	58	421
4	42	45	75	85	97	369
5	62	59	41	86	91	626
6	61	24	24	81	85	597
7	3	71	3	93	30	780
8	97	34	36	31	38	450

Weightage for make-span and tardiness is same i.e 0.5 (W_1=0.5, W_2=0.5).

Table 2. Optimum Sequence and Make-span for Case Study

S. No	No. of iteration	Time taken for simulation	Make-span	Total tardiness	Tardiness jobs	Fitness value	Job sequence
1	50	54sec	698	24	3	361	3-6-8-2-5-7-1-4
2	75	1min 40sec	698	24	3	361	3-6-8-2-5-7-1-4
3	100	2min 46sec	698	24	3	361	7-6-4-8-5-1-3-2
4	150	4min 35sec	695	4	4	349.5	8-6-5-2-4-1-3-7

Table 2 shows the evolution of fitness value with the four simulation runs. The best fitness value and sequence of jobs among four simulation runs are 349.5 and 8-6-5-2-4-1-3-7 respectively. And corresponding sequence make-span and tardiness are 695 and 4 respectively.

Case Study-2

It is an 8 Jobs and 8 Machines General Flow Shop problem. This is a typical flow shop problem where some jobs require few machines available in the shop and other jobs require some other machines available on shop. Table 3 shows the processing time of various jobs on the machines and each job due date. A zero entry in a cell indicates that a job does not require a particular machine.

Table 3. Processing Time of Jobs on Machine and Job Due Date (All values are in minutes**)**

Job	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	Due Date
1	20	0	18	12	22	16	0	18	180
2	18	22	20	0	16	0	15	20	200
3	0	12	20	20	0	18	17	13	210
4	14	19	0	17	20	14	26	0	128
5	18	0	21	0	16	10	20	14	170
6	0	15	14	20	23	0	15	20	140
7	21	21	0	10	10	23	23	0	146
8	13	22	19	13	0	17	0	12	150

Weightage for make-span and tardiness is same i.e 0.5 (W_1=0.5, W_2=0.5).

Table 4. Optimum Sequence and Make-span for Case Study

S. No	No. of iteration	Time taken for simulation	Make-span	Total tardiness	Tardiness jobs	Fitness value	Job sequence
1	50	50sec	234	73	5	155.5	3-7-2-1-6-5-8-4
2	75	1min 41sec	214	92	4	153	6-1-7-3-8-2-5-4
3	100	3min 30sec	214	92	4	153	8-1-3-4-6-5-2-7
4	150	4min 45sec	214	92	4	153	8-1-3-4-6-5-2-7

Table 4 shows the evolution of fitness value with the four simulation runs. The best fitness value and sequence of jobs among four simulation runs are 153 and 6-1-7-3-8-2-5-4 respectively. And corresponding sequence make-span and tardiness are 214 and 92 respectively.

CONCLUSIONS

In present work an attempt has been made to solve multi objective flow shop scheduling problem using Genetic Algorithm. The ease of JAVA program in UI (User Interface) handling and OOP (Object Oriented Programming) concepts are utilized to incorporate GA in effective manner. The developed Java program is able to answer the variety of scheduling problems. Two case studies were applied to verify the effectiveness of the Genetic Algorithm. The best schedule plans which have the minimum fitness value of (make-span and total tardiness) at each generation are presented. Increasing the number of generations when a little change in the fitness value exists results in increase in time taken to complete the generation.

REFERENCES:

[1] Ashwani Dhingra, Pankaj Chandna, "Multi-objective Flow Shop Scheduling using Hybrid Simulated Annealing", Measuring Business Excellence, Vol. 14, No. 3, pp. 30 – 41, 2010.
[2] Ashwani Dhingra, Pankaj Chandna "A Bi-criteria M-Machine SDSTFlow Shop Scheduling Using Modified Heuristic Genetic Algorithm", International Journal Engineering Science & Technology, Volume 2, No.5, pp. 216-225, 2010.
[3] Budi Santosa, Ainur Rofiq "The Development of Simulated Annealing Algorithm for Hybrid Flow Shop Scheduling Problem to Minimize Makespan and TotalTardiness", International Conference on Industrial Engineering and Operations Management Bali, Indonesia, January 7 – 9, 2014.
[4] Cengiz Kahraman, Orhan Engin, Ihsan Kaya, "An Application of Effective Genetic Algorithms For Solving Hybrid Flow Shop Scheduling Problems", International Journal of Computational Intelligence Systems, Volume1, No. 2, pp.134-147, May, 2008.
[5] Chia-Shin Chung, James Flynn, Walter Rom, PiotrStaliński "A Genetic Algorithm to Minimize the Total Tardiness for M-Machine Permutation Flow shop Problems", Entrepreneurship, Management and Innovation , Volume 8, Issue 2, pp.26-43, 2012.
[6] Colin R. Rreeves, "A Genetic Algorithm for Flow Shop Sequence" Computer operations Research, Volume 22, pp.5-13, 1995.
[7] Gaurav Kumar, Shailja Singhal, "Genetic Algorithm Optimization of Flow Shop Scheduling Problem with Sequence Dependent Setup time and Lot Splitting" International Association of Scientific Innovation and Research, 4 (1), pp.62-71, March-May, 2013.
[8] Godfrey C. Onwubolu, and Michael Mutingi "Genetic Algorithm for Minimizing Tardiness in Fow-Shop Scheduling"Production Planning & Control, Volume 10, NO. 5,pp. 462–471, November 2010.
[9] HelaBoukef, Mohamed Benrejeb, Pierre Borne "A Proposed Genetic Algorithm Coding for Flow-Shop SchedulingProblems"International Journal of Computers, Communications & ControlVolumeII, No. 3, pp. 229-240, 2007.
[10] NeelamTyagi and VarshneyR.G "A Model to Study Genetic Algorithm for the FlowShop Scheduling Problem"Journal of Information and Operations Management, Volume 3, Issue 1, pp.38-42, January 15, 2012.
[11] Imran Ali Chaudhry, Abdul Munem Khan "Minimizing make span for a no-wait flow shop using genetic algorithm", Indian Academy of Sciences, Volume. 37, Part 6, pp. 695–707, December 2012.
[12] Rahimi-Vahed, R., Mirghorbani, S. M, "A multi-objective particles warm for a flow shop scheduling problem", Journal of Combinatorial Optimization 13(1), pp.79–102, 2007.
[13] Sadegheih .A, "Scheduling problem using genetic algorithm, simulated annealing and the effects of parameter values on GA performance", Applied Mathematical Modeling, pp.147–154, March2005.
[14] ShahsavariPour .N, M.H. Abolhasani Ashkezari, H. Mohammadi Andargoli, "A Genetic Algorithm Coupled with Tabu Search for Bi-Objective Permutation Flow Shop", International Journal of Research in Industrial Engineering, Volume 2, Number 1, pp-20-29, January 2013