ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

WORKLOAD BALANCING IN IDENTICAL PARALLEL MACHINE SCHEDULING WHILE PLANNING IN FLEXIBLE MANUFACTURING SYSTEM USING GENETIC ALGORITHM

B. V. Raghavendra[1] and A. N. N. Murthy[2]
[1]Department of Mechanical Engineering, JSS Academy of Technical Education, Bangalore, India
[2]T. John Institute of Technology, Bangalore, India
E-Mail: bv_raaghu@yahoo.co.in

ABSTRACT

This paper addresses the loading problem in flexible manufacturing systems (FMSs). The problem involves the assignment of the operation or jobs to the identical parallel machine to process required part types that have been selected to be produced simultaneously. A genetic algorithm (GA) based heuristic approach is proposed in this paper for minimizing the imbalance of workload among the identical parallel machines. The program is coded in MATLAB and conducted the trials on compatible IBM/PC. Computational results are presented in appendix and compared for different test problems to demonstrate the efficiency and effectiveness of the suggested procedure.

Keywords: workload balance, FMS, parallel machine, genetic algorithm.

1. INTRODUCTION

A flexible manufacturing system (FMS) is a production system that consists of numerically controlled machines connected via an automated transportation system. Every process within an FMS is controlled by a central computer. An FMS has the capability to process parts of a certain part spectrum in arbitrary order.

Depending on the operation to be performed by the machine, FMSs can consist of identical and different machines (Tempelmeier and Huhn, 1993). Identical machines can perform identical operations. If they are equipped with identical tools, they can be used alternatively for an operation and thus offer a choice in the processing of a part type [1].

Identical parallel machine scheduling problem for minimizing the imbalance of workload between the machines and the makespan has been solved by operational methods such as dynamic programming, branch and bound method, integer programming etc. These methods can give an optimal solution for a reasonably sized problem, however, in the case of a large scale problem these methods have limitation of applications of mathematical optimization techniques, Heuristic procedure [2] is suitable for identical parallel machine scheduling problem of small scale, but in case of processing objects of larger scale, heuristic procedure is not yet effective enough, especially the accuracy of the solution need improving. Genetic algorithm (GA) [3] has been applied in those fields such as combinatorial optimization successfully in view of it characteristic such as near optimization, high speed, and easy realization.

The aim of machine scheduling is to assign jobs to the machines based on some related objective function. Different approaches are being followed to solve machine-scheduling problems. As the resources are limited, it is essential to devise an optimal schedule in a manufacturing environment to increase the productivity. Multiple machines are used in parallel for processing the jobs to meet the demand. In parallel machine scheduling, there are 'm' machines to which 'n' jobs are assigned satisfying the precedence constraints based on an objective function. Workload balancing among the machines is one of the surrogate objectives considered for solving this type of scheduling problems. Workload balancing in a shop floor helps to reduce work-in-process (WIP) inventory, makespan, increase the throughput, and machine utilization. It removes the bottleneck present along the product line. The machine with less workload is selected for assigning a new job from the lists of jobs [4].

In this paper the author has worked out heuristic based GA approach and the results of the GA approach is compared with the methods proposed in the paper [1, 3] and found that proposed GA methods gives better results.

2. PROBLEM FORMULATION

In this study, the identical parallel machine scheduling problem for minimizing the unbalance between the machines or minimizing the makespan is defined as follows: there are 'n' independent jobs and 'm' identical machines; each job has its fixed processing time. The processing job can be completed by either of the machines. We want to find out the sequence and the assignment of the job with the operation on the machine so that workload imbalance among machines will be minimized thereby minimizing the makespan.

A heuristic procedure is explained for determining the fitness (Objective) function for selected part type's sequence which needs to be loaded on the parallel machines. There are n! Ways of sequence for 'n' different part types. GA is applied for determining the best sequence so that the objective function is met (i.e. minimizing the imbalance among the parallel machines there by shorter or reducing the makespan).

Assumptions

a) Job Loading on 'm' Parallel Machines and less than or equal to the part types.
b) Simultaneously processing of selected part types.

c) Machine Capacity is sufficiently available and enough tool slots are available on machine to process selected part type.
d) All the jobs and machines are simultaneously available in the beginning.
e) Transportation time required to move a job/tool on the machine is negligible.
f) Time required loading the tool and set-up of the machine is not considered.
g) Constraints related to material handling system, availability of other resources such as pallets, fixtures are relaxed.
h) Required tools will be loaded on machine for every part types and set-up,
i) The set-up ($S_{i)}$ will be the precedence constraint for the next set-up ($S_{i+1)}$. The set-up precedence constraint should be satisfied.

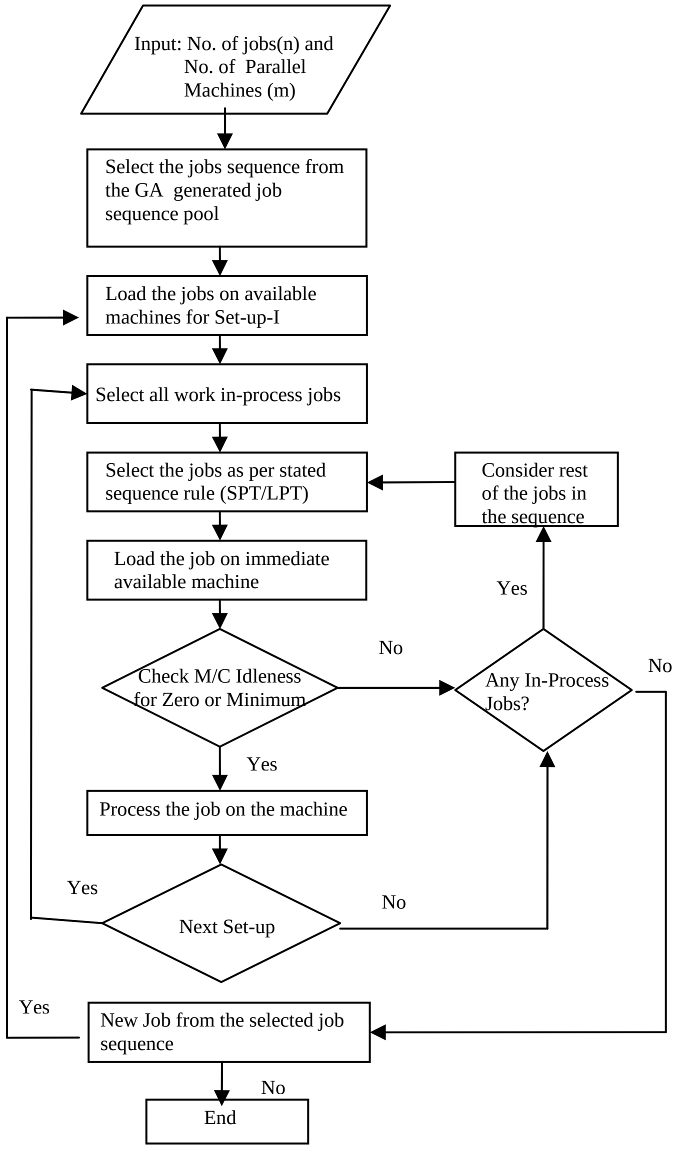

Figure-1. Flow chart for objective function (Fitness Function).

3. GA: I SET-UP BASED ON GA GENERATED JOB SEQUENCE AND REMAINING SET-UPS BASED ON SPT/LPT RULE

a) For I set-up job sequence is based on GA generated job sequence
b) For rest of the set-up the sequence rule is based on SPT/LPT respectively considering machining time of immediate competing set-up of the work-in-process jobs (WIP).

3.1. Procedure (refer Figure-1)

a) At the beginning all the machines are available and select the job to load on the machine based on GA generated job sequence.
b) If machine is free load the machine by WIP jobs. If more than one WIP jobs are waiting to process then select the job which has SPT/LPT in the rest of the set-ups respectively. But machine should not be idle

VOL. 6, NO. 1, JANUARY 2011 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

while loading the jobs. If so then select the next WIP job so that machine is not idle.

c) If no WIP jobs are available then load the machine with the remaining jobs from the randomly selected job pool.
d) Repeat above steps till completing all the jobs.

4. GA IN PARALLEL MACHINE SCHEDULING

In parallel machines all the machines are identical and the job can be processed on any machines. The GA is used to compute the relative percentage of imbalance in workloads among the machines. The designing of GA requires chromosomal representation, type of initial population, evaluation of each chromosome for objective function, genetic operators such as population size, crossover, mutation, reproduction and stopping criteria [5] .Representation (encoding) plays a major role in the development of GAs. We have decided to use the sequence-oriented representation scheme. The initial population is generated randomly [6]. The numbers (Gens) is the chromosomes are jobs. In this research relative percentage index ((Maximum workload-Minimum workload)/Maximum workload) is used to evaluation of the each chromosome. The chromosome (Sequence) which has the least relative percentage index is the best sequence which will be the optimum or near to the optimum. The workload of a machine is the sum of processing time of all jobs allotted to it. Since the workload balance is the combinatorial optimization problem, hence partially mapped cross over is used. The mutation operator used in this research is similar to reciprocal exchange, which randomly selects two positions in the string and swaps the part types in these two positions to generate a new string. Termination is the criterion by which the genetic algorithm decides whether to continue searching or stop the search. In the Genetic algorithm following types of termination can be applied. Generation Number (A termination method that stops the evolution when the user specified maximum number of evolution has been run. This termination method is always active.), Evolution time, Fitness threshold, Fitness Convergence, Population Convergence, Gene Convergence. In this research Generation Number is used for termination.

The most difficult and time consuming issue in the successful operation of GAs determining good parameter settings. These parameters are crossover probability (PC), mutation probability (PM), population size (POP_SIZE), number of generation (MAX_GEN), etc. However, Michalewicz (1992) mentioned that the determination of proper values of these genetic parameters still remains an art rather than science.

5. RESULTS AND DISCUSSIONS

In this paper six test problems are worked out with setting the GA parameter as population size 20, best parents 12, maximum generation 50 and crossover fraction 0.9. The performance and result of proposed method of GA is compared with the methods proposed in the reference [1] for different data sets. Data Set 1(29 operation 3 identical machines), Data Set 2 (33 operation 5 identical machines),

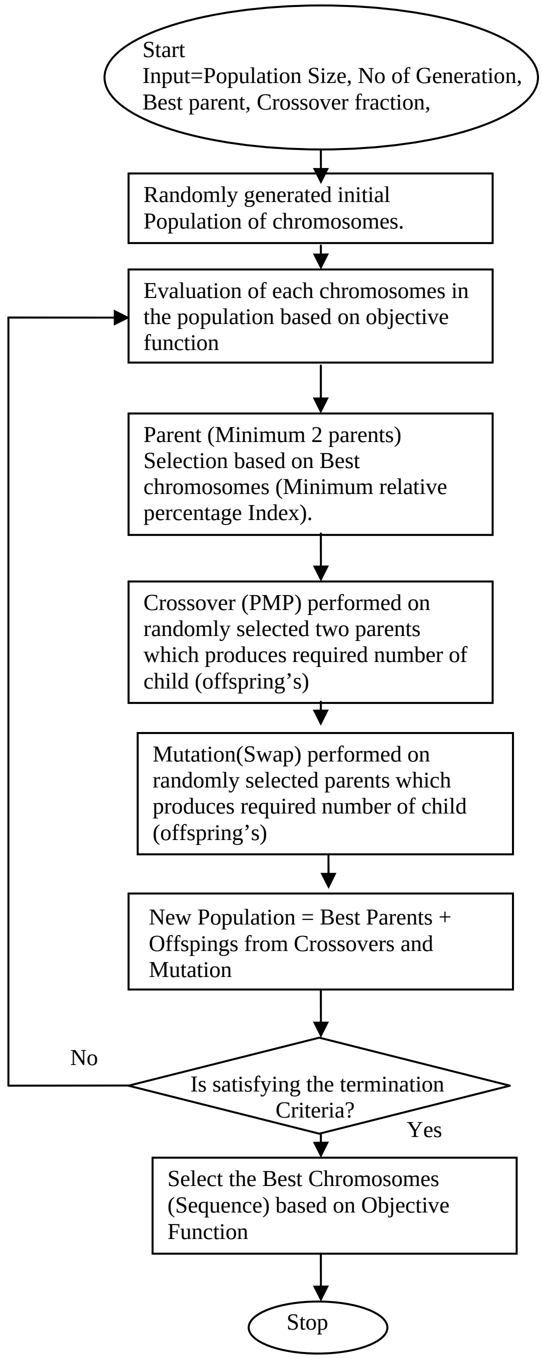

Figure-2. Flow chart of genetic algorithm.

Data Set 3 (21 operation 6 identical machines). The results of machine imbalances in all the three data sets is encouraging compare to STA, SYSR and IE methods proposed by Heinrich Kuhn. However the results of IE and proposed GA methods for the Data Set 2 are same.

The performance and result of proposed method of GA is also compared with the methods proposed by Liu Min, Wu Cheng (1999) [3]. The makespan for the data set

VOL. 6, NO. 1, JANUARY 2011 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

4(7 jobs 3 parallel machines) and data set 5 (10 jobs 2 parallel machines) are same as that of SA and GA methods. However in the Data Set: 6 (30 jobs 10 parallel machines) the makespan of proposed GA result is superior to the other methods proposed by Liu Min, Wu Cheng (1999) [3]. The imbalance between the identical machine is also compared for the data Set: 6 and found that the proposed GA method shows reducing the imbalance between the identical machines.

6. CONCLUSIONS

In this paper GA based heuristic procedure for the loading problems in FMS in Identical parallel machine is presented. The GA based heuristic procedure is compared with the heuristic procedure proposed in the paper [1, 3] for minimizing the workload unbalance between the machines and reducing the makespan. However proposed GA based heuristic procedure gives better results in view of accuracy and the quick solution. This will help to finding the better sequence of part types within reasonable time in real time application to meet the objective function of workload balance thereby reducing makespan, work-in-process inventory and increase machine utilization. The future research can be extended to consider other resources such as pallets, AGV, Robots etc.

7. APPENDIX

The appendix gives the detailed data sets and solutions. First three test problems data sets of first Table gives the input data for operation number and processing time, second table gives the solution for machine loading with the operations allotted for different approaches. Third Table of these data sets gives the work load of each machine for different approaches. The next three test problems data sets of first table gives input data for job number and the processing time, second table gives the solution for minimum makespan, last data set of third table is the job allocation for each machine and fourth table is the total workload allocation for each machine.

Data Set: 1 (29 Operations, 3 machines)

Table-1. (Input data).

Operations	Proce.Time	Operations	Proce.Time	Operations	Proce.Time	Operations	Proce.Time	Operations	Proce. Time	Operations	Proce. Time
1	14	6	3	11	11	16	1	21	14	26	23
2	16	7	11	12	25	17	26	22	20	27	21
3	26	8	25	13	21	18	18	23	5	28	17
4	3	9	2	14	14	19	13	24	12	29	19
5	25	10	24	15	25	20	2	25	21		

Table-2. Solution for operation allocation details machine wise for data set: 1.

Operations	Heinrich Kuhn 1995			*GA	Operations	Heinrich Kuhn 1995			*GA	Operations	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE			STA	SYSR	IE			STA	SYSR	IE	
1	1	1	3	3	11	1	1	2	1	21	2	2	3	2
2	2	2	3	1	12	1	1	1	3	22	3	3	3	1
3	3	3	3	3	13	1	1	2	3	23	2	2	2	1
4	3	1	2	1	14	3	3	3	1	24	2	2	3	2
5	1	1	1	1	15	3	3	1	2	25	2	2	1	2
6	1	3	1	2	16	3	3	1	1	26	2	3	2	1
7	1	1	2	2	17	3	2	2	3	27	2	2	2	3
8	1	1	1	2	18	3	3	3	2	28	2	2	3	1
9	1	1	1	1	19	3	1	2	1	29	2	2	2	3
10	1	3	1	2	20	3	2	1	1					

Table-3. Machine work load for data set: 1.

Machine	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE	
Machine-1	161	150	153	152
Machine-2	148	153	153	153
Machine-3	148	154	151	152

VOL. 6, NO. 1, JANUARY 2011 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Data Set: 2. (33 Operations, 5 machines)

Table-4. (Input data).

Operations	Proce.Time
1	23
2	29
3	7
4	2
5	25
6	10
7	14
8	3
9	26
10	23
11	10
12	7
13	30
14	25
15	1
16	8
17	18
18	13
19	15
20	28
21	30
22	14
23	28
24	14
25	20
26	18
27	27
28	19
29	11
30	25
31	5
32	22
33	30

Table-5. Solution for operation allocation details machine wise for data set: 2.

Operations	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE	
1	1	5	3	1
2	2	2	2	3
3	3	3	2	5
4	4	3	4	5
5	5	4	2	4
6	1	2	4	2
7	1	1	5	4
8	1	1	3	2
9	1	1	1	5
10	1	3	3	3
11	3	3	5	5
12	3	3	1	2
13	3	3	5	2
14	3	1	2	3
15	3	3	3	1
16	2	2	4	5
17	3	4	4	2
18	2	5	4	2
19	2	2	4	5
20	2	2	1	1
21	2	3	4	2
22	5	5	5	4
23	2	2	1	5
24	5	5	3	3
25	5	1	4	5
26	5	5	5	1
27	5	1	1	1
28	4	4	5	1
29	4	5	5	4
30	5	5	2	3
31	4	4	2	2
32	4	4	3	4
33	4	4	3	4

Table-6. Machine work load for data set: 2.

Machine	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE	
Machine-1	99	115	116	116
Machine-2	151	118	116	116
Machine-3	98	110	116	116
Machine-4	89	119	116	116
Machine-5	143	118	116	116

Data Set: 3. (21 Operations, 6 machines)

Table-7. (Input data).

Operations.	Proce.Time
1	23
2	29
3	21
4	11
5	20
6	28
7	10
8	18
9	1
10	6
11	28
12	19
13	5
14	6
15	28
16	27
17	8
18	7
19	29
20	10
21	7

VOL. 6, NO. 1, JANUARY 2011 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-8. Solution for operation allocation details machine wise for data set: 3.

Operations	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE	
1	1	4	1	1
2	2	1	6	6
3	3	4	2	5
4	4	5	6	1
5	5	5	3	4
6	6	6	2	2
7	5	2	3	4
8	5	5	4	2
9	2	2	2	3
10	2	3	4	5
11	6	2	4	3
12	1	3	5	4
13	1	1	5	2
14	2	6	4	1
15	6	6	3	3
16	1	1	2	6
17	4	4	2	1
18	4	4	1	4
19	3	3	5	5
20	3	2	6	1
21	3	3	6	2

Table-9. Machine work load for data set: 3.

Machine	Heinrich Kuhn 1995			*GA
	STA	SYSR	IE	
Machine-1	74	61	57	58
Machine-2	42	49	58	58
Machine-3	60	61	58	57
Machine-4	33	59	58	56
Machine-5	48	49	53	56
Machine-6	84	62	57	56

Data Set: 4 (7 Jobs, 3 machines)

Table-10. (Input data).

Job No.	Proce.Time
1	6
2	6
3	4
4	4
5	4
6	3
7	3

Table-11. Solution for minimizing the makespan for data set: 4.

Problem Scale	Liu Min, Wu Cheng (1999)			*GA
	LPT	SA	GA	
7 Jobs X 3 Machines	11	10	10	10

Data Set: 5 (10 Jobs, 2 machines)

Table-12. (Input data).

Job No.	Proce.Time
1	3
2	2
3	6
4	4
5	5
6	7
7	8
8	6
9	2
10	6

Table-13. Solution for minimizing the makespan for data set: 5.

Problem Scale	Liu Min, Wu Cheng (1999)		*GA
	SA	GA	
10 Jobs X 2 Machines	25	25	25

VOL. 6, NO. 1, JANUARY 2011 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Data Set: 6 (30 Jobs, 10 machines)

Table-14. (Input data).

Job No.	Proce. Time	Job No.	Proce. Time	Job No.	Proce. Time	Job No.	Proce. Time
1	3	9	4	17	14	25	7
2	2	10	12	18	6	26	23
3	6	11	10	19	17	27	15
4	4	12	8	20	27	28	18
5	5	13	22	21	11	29	15
6	7	14	11	22	17	30	13
7	9	15	8	23	26		
8	13	16	26	24	16		

Table-15. Solution for minimizing the makespan for data set: 6.

Problem Scale	Liu Min, Wu Cheng (1999)		*GA
	SA	GA	
30 Jobs X 10 Machines	44	41	39

Table-16. Machine loading for data set: 6.

Job No.	Machine No		Job No.	Machine No		Job No.	Machine No		Job No.	Machine No	
	GA	*GA		GA	*GA		GA	*GA		GA	*GA
1	8	7	9	9	7	17	2	4	25	10	8
2	10	7	10	10	8	18	4	5	26	3	9
3	3	1	11	10	6	19	5	10	27	7	5
4	2	10	12	1	3	20	4	2	28	8	5
5	1	3	13	2	1	21	9	4	29	7	10
6	4	7	14	7	2	22	1	8	30	8	7
7	10	1	15	3	7	23	9	3			
8	6	4	16	6	6	24	5	9			

Table-17. Machine work load for data set: 6.

Method	M/c:1	M/c:2	M/c:3	M/c:4	M/c:5	M/c:6	M/c:7	M/c:8	M/c:9	M/c:10
GA	30	40	37	40	33	39	41	34	41	40
*GA	37	38	39	38	39	36	37	36	39	36

*GA ……….... GA based heuristic method proposed in this paper
Proce. Time….Processing time
M/c…………..Machines

REFERENCES

[1] Heinrich Kuhn. 1995. A Heuristic Algorithm for the Loading Problem in Flexible Manufacturing System. The International Journal of FMS. 7: 229-254.

[2] Blackstone HG, Phillips ET, Hogg GL. 1982. The state-of-art survey of dispatching rules for manufacturing job shop operations. International Journal of Production research. 20(1): 27-45.

[3] Liu Min, Wu Cheng. 1999. A genetic algorithm for minimizing the makespan in the case of scheduling identical parallel machines, Artificial Intelligence in Engineering. 13: 399-403.

[4] S. Rajakumar, V.P. Arunachalam, V. Selladurai. 2006. Workflow balancing in parallel machine scheduling with precedence constraints using genetic algorithm. Journal of Manufacturing Technology Management. 17(2): 239-254.

[5] M.E. Kurz and R.g. Asking. 2001. Heuristic scheduling of parallel machines with sequence-dependent set-up times. Intl J. Prod. Res. 39(16): 3747-3769.

[6] Mahanim Omar, Adam Baharum, Hahya Abu Hasan. 2006. A Job- Shop Scheduling problem (JSSP) using Genetic Algorithm (GA). Proceedings of the 2nd IMT-GT Regional Conference on Mathematics, Statistics and Applicaions University Sains Malaysia, Penang, June 13-15.

[7] B.V. Raghavendra, Dr. A.N.N. Murthy. 2010. Some solution approaches to reduce the Imbalance of Workload in Parallel Machines while Planning in Flexible Manufacturing System. International Journal of Engineering Science and Technology. 2(5): 724-730.