*International Journal of Research in Advent Technology, Vol.3, No.9, September 2015*
*E-ISSN: 2321-9637*
*Available online at www.ijrat.org*

# Job Shop Scheduling With Alternate Process Plan by Using Genetic Algorithm

Randhir Kumar[1], Vijay Pandey[2]
*Department of Mechanical Engineering [1], Bengal College of Engineering and Technology, Durgapur, West Bengal, India*
*Department of Production Engineering[2], Birla Institute of Technology, Mesra, Ranchi, Jharkhand, India*
*Email: randhir1849@gmail.com[1], vpandeybit@rediffmail.com [2]*

**Abstract-** Scheduling in a job-shop system is a challenging task, however it costly and time consuming. Successful implementation of automated manufacturing system highly depends on effective utilization of resource. Efficient scheduling algorithm for alternate process plan may increase the throughput rate, utilization of machine and guarantee a reasonable return of investment. In this paper, the objective is to prepare an alternate model with minimum makespan value by using genetic algorithm. For this makespan value to solve the various job sequencing problem, utilization of machine, cost of machine for a production shop that is characterized by alternate routing and flexible machines also investigates an optimization for scheduling job in a just-in- time environment. All jobs can be processed through alternate routing to be processed in a specified order of operation. Each operation has to be processed on one of a set of resources (e.g. machine) with possibly different efficiency and hence processing time. The objective is to minimize the make span of the job, mean flow time, utilization of machine, Average utilization of machine.

**Index Terms-** *J*ob shop; Production scheduling; Genetic algorithms.

## 1. INTRODUCTION

Job shop scheduling problem is a typical combinatorial optimization problem, in job shop scheduling problem each and every job is not processed through all machines in the same sequence as in flow shop scheduling problem. Here different jobs have different sequence of operations and jobs may or may not pass through every machine and each machine has different sequence of jobs.

The production scheduling and alternate process planning is an important decision making n operation level. Many of the scheduling problem arises, that very hard to solve due to complex nature of problem. In this paper interested in the scheduling problem on the environment, where several alternatives machine and job consists a chain of operations to be processed, this is known to job shop problem (JSP). This problem is well known to machine scheduling problem. It contains set of machine and set of job, consists set of operation that can be processed in more than one machine with different processing time and cost. This is called alternative routing. This alternative operation could be used when if one machine tool is temporarily overloaded while another is idle. Each machine is capable of performing more than one type of operation. For a given operation there must be one machine is capable of performing it. This problem of scheduling is decomposing in to two sub problem. The routing sub problem, which assigning each operation to a machine out of alternative set of machine and scheduling sub problem, which consists of sequence the assign operation on all the machine in order to obtain the objective function. This proper representation plays a key role in the development of a genetic algorithm.

## 2. BACKGROUND

Wihelm and Shin [1] they study to investigate the effectiveness of alternate operation in flexible manufacturing system (FMS), In addition computational experiment are performed to reduce the flow time while increasing machine utilization. Hakins et al. [2] considered alternate machine tool routing to improve productivity, and alternative machine results in reducing lead time thus overall improvement in machine utilization. Nasr and Elsayed [3] emphasizes on machine assignment and task scheduling and simultaneously machine utilization and expediting the flow of work piece. Liang and Dutta [4] purposed a mixed integer programming formulation for simultaneously process planning and machine loading, also formulate optimal process plan for each part and each machine by keeping cost in mind. Hutchison at al. [5] gives an optimal solution procedure to investigate the effect of routing flexibility on job shop FMS. Dauzere and Paulli [6] introduced two local search heuristics to solve job shop problem with alternate routing in order to minimize makespan time. Kim and Egbelu [7] developed a mixed integer programming model for scheduling a set of jobs through a shop when each job is supplied or provided with multiple process plan or process routing. Candido et al. [8] gives alternate routes by dividing the routes into sub processes, each process similar sequence of operation by use of genetic algorithm (GA) to minimize makespan. Hussain and Joshi [9] proposed two way pass for job shop scheduling problem (JSSP) by use of GA. The first pass picks the alternative using GA, and second pass provide the order and start time of job on the selected alternative by solving a non-linear program. Jawahar et al. [10] gives a search heuristic for scheduling FMS with alternate routing. The heuristic first select randomly a route for each operation and then a genetic based algorithms, evolves a priority dispatching rule for each machine to resolve the conflicts that arise in the schedule generation procedure. Kim and Egbelu [11] introduced two local search heuristics to solve job shop problem with alternate routing in order to minimize makespan time. Chen et al. [12] used GA with new chromosome representation to solve flexible job shop problems. Beck and Fox [13] addressed a constraint-directed scheduling technique to deal with the case where the scheduling problem includes alternative activities. Thomalla [14] introduced Lagrangian relaxation for job shop scheduling problem with alternate route to minimize the sum of the weighted quadratic tardiness of the job. Zhou et al. [15] purposed hybrid GA for job shop scheduling problem. They used priority rule such as short processing time for genetic search to devise hybrid genetic algorithms. Choi and Choi [16] considered alternative routes and sequence dependent setup together to minimize makespan. They developed a mathematical programming model and local search algorithms using dispatching rules. Kacem et al. [17] used job sequencing list coding scheme and developed an approach by localization to find promising initial assignment. Kim et al. [18] proposed symbiotic evolutionary algorithms for job shop FMS integrated with process planning. They considered two chromosomes of different lengths, one is involved in operation sequence for completion of a job and other is responsible for machine assignment. Kim et al [19] purposed an asymmetric multi-leveled symbiotic evolutionary algorithm, and applied it to a integrated problem of process planning and scheduling in FMS. Zhang and Zen [20] gives multistage based GA to solve job shop flexible manufacturing system. Tavakkoli et al. [21] developed a new mathematical programming model for a multi-criteria parallel machine scheduling problem. The aim was to minimize the total earliness/tardiness penalties and machine cost simultaneously. Demir [22] considered scheduling and process planning function concurrently and proved a benefit of flexible process planning and integration, also they compared random search, genetic search and hybrid search technique with ordinary solutions. Mehrabad and Fattahi [23] presented a tabu search algorithm that solves the flexible job shop scheduling to minimize the makespan time. Zhang et al. [24] introduced a hybrid of ant colony and particle swarm optimization to solve flexible JSSP. Pezzella et al. [25] integrate different strategies for generating the initial population, selecting new individual for reproduction and reproducing new individuals. Xing et al. [26] introduced a knowledge based heuristic algorithm, combined with empirical knowledge for multi-objective flexible JSSP. Xing et al. [27] also presented a simulation model to solve a multi-objective flexible JSSP. Kim and Egbelu [28] proposed a mathematical approach to solve modelled problems. They gives two algorithms, one of the pre-processing algorithm, finds the optimal solution and the second iterative algorithms, is also effective in finding good solution. Chaudhry [29] presented a domain independent GA in spread sheet environment approach for the JSSP with alternative machine.

## 3. PROBLEM AND ASSUMPTIONS

In real time manufacturing system many challenges of the demand of high variety and low volume product is there. Therefore, it is a needed to schedule a process plan such that to achieve the objectives. The schedule of job must improve the flexibility that handles less impact on the system.

The problem is to schedule the job with alternate route of processes in which each job takes alternate machine and processing time. It is very difficult to schedule the job among these machines because the system performance is based on the deviation of completion time. These schedules have the different mean flow, utilization of machine, average utilization of machine so according to that which one is suited to the requirement. The requirement may be the time, cost, machine utilization etc. Assumption that has tried to meet certain objective is:

- Machines are available all the time. (No machine interruption or down time).
- Jobs are available at t=0. (Jobs don't come late).
- No job splitting. (Entire task complete on particular machine).
- No job interruptions. (Once taken complete up then process to next).

- Processing time is known also fixed and known corresponding to the operations for each job.
- Each machine can process at most one operation at a time.
- A given operations be able to performed by one or more non-identical machines (called alternative machines).
- No priorities are assigned for any job or operation.
- Breakdowns are not considered.

### 3.1. *Objectives*

The main objective of this research is:

1. Use genetic algorithms to develop an alternate process plan.
2. To generate multiple ways to sequence the operations required to realize the finished product.
3. The alternate process plan schedule in such a way that it takes the minimum time to complete for the entire job or make span.
4. Min make span = Min [Max ($C_{1J1}$ ,$C_{2J2}$ . . . , $C_{nJn}$)]
5. Minimize mean flow time
   *Mean flow time* = $\frac{1}{n}\sum_{i=1}^{n}\sum_{j=ji} Cij$
6. Maximum average resource utilization: The percentage utilization of an individual machine is calculated based on the maximum flow time .i.e for a schedule the individual and average resource utilization is calculated as follows:
   *Utilization* = $U_k$ = $\frac{\text{total busy time}}{\max(Ci)}$
7. Average utilization= $\frac{1}{m}\sum_{k=1}^{m} Uk$

## 4. METHODOLOGY

### 4.1. *Genetic Algorithms*

Genetic Algorithms (GAs) were developed by Prof. John Holland and his students at the University of Michigan during the 1960s and 1970s. Essentially, they are a method of "breeding" computer programs and solutions to optimization or search problems by means of simulated evolution. The evolutionary algorithms use the three main principles of the natural evolution: reproduction, natural selection and diversity of the species, maintained by the differences of each generation with the previous. The GA represents an intelligent exploitation of random search use to solve optimization problem. Although randomize, GA are no means random instead they are exploit the historical information to direct the search in to region of better performance within the search space ,this basic technique to design to simulate process of natural system necessary for evolution specially follow the principle of Charles Darwin “survival the fittest”.

### *4.2 Methodology Used In Application Of Genetic Algorithm*

#### *4.2.1 Initialization*

GA is a search technique that starts with an initial set of (random) solution called population. Each individual in the population is called chromosome which may represent the solution of the problem. The chromosome evolves through successive iteration called generation that may be more fitted or may not be. Initially chromosome as composed of two parts. The first part is for the assignment of alternative machines, and the second part is the relative processing order between jobs (see chromosome). The length of each chromosome is equal to the total number of operations.

| Assignment chromosome | Sequence chromosome |
|---|---|
| $O_{11}$....$O_{1Ji}$....$O_{n1}$....$O_{nji}$ | $O_{11}$....$O_{1Ji}$....$O_{n1}$....$O_{nji}$ |

Fig. 1: Structure of chromosome.

#### *4.2.2. Selection*

During each successive generation, a population of the existing population is selected to reproduce a new generation. Each individual is given a fitness value, and those which have higher fitness value are selected while those which have lower fitness value are discarded. Selection is an important step because the following results depend on the generation being selected the very first step.

#### *4.2.3. Crossover and Mutation*

In general, the crossover operator is regarded as main genetic operator on two chromosomes at a time and generates offspring by combining features of both chromosomes. Crossover is performed with a probability between two selected individuals by exchanging parts of it to form two new individuals called offspring.

For the machine assignment, i.e., first six genes, single point crossover are implemented. Now for the order of the job, i.e., order crossover is performed, then a mutation takes place; mutation is intended to prevent falling of a solution in the population in to a local optimum of the solved problem.

### *4.2.4. Fitness Function*

Fitness is a measure of how well an algorithm has learnt to predict the outputs from the inputs. In this case the number of jobs and their processing time are known, fixed and assumed that there are no machines breakdown occurred. For each chromosome the fitness function aims to find the minimum Cmax, and is represented as follows:

Min [Max (C1J1 ,C2J2 . . . , CnJn)]

Where, J = index of operation (j=1,...........,Ji)

I = job number i (i= 1,............., n)

K = machine k (k=1,2,...........m)

ji= number of operation required to complete job i.

Ojj= Operation number j of job i ($O_{i1}$, $O_{i2}$,... $O_{iji}$)

Cij= completion time of operation Oij.

### *4.2.5. Termination*

The genetic algorithm stops when the specified number of generations has evolved.

### ***4.3 Development of model for Scheduling***

In this model, six genes are initially taken for the two jobs only for testing purpose. The length of these chromosomes is equal to the total number of operation. This research also apply selection mechanism breed a new generation. Individual solutions are selected through a fitness based process. The step is to generate new generation by the use of crossover and mutation; actually the different crossover and mutation are applied for the machine assignment and sequence chromosome. For each new population to be produced, that may be the solution of problem. But if further more suited value is required the cycle is again repeated for further new generation. This generation process is repeated until the termination condition has been reached. These termination criteria may be the number of cycle; objective achieved, or times limits etc. The flowchart for the development of model is given in fig. 2. In this case, run the program using MATLAB 2012, and achieved makespan of the process plan and its other objective are to be obtained for these schedules on the machine. These objectives are related to the mean flow time, utilization of machine, average utilization of machine etc. The different value of these objective are to be achieved for the same makespan value , thus which one value of objective are required to give the most priority.

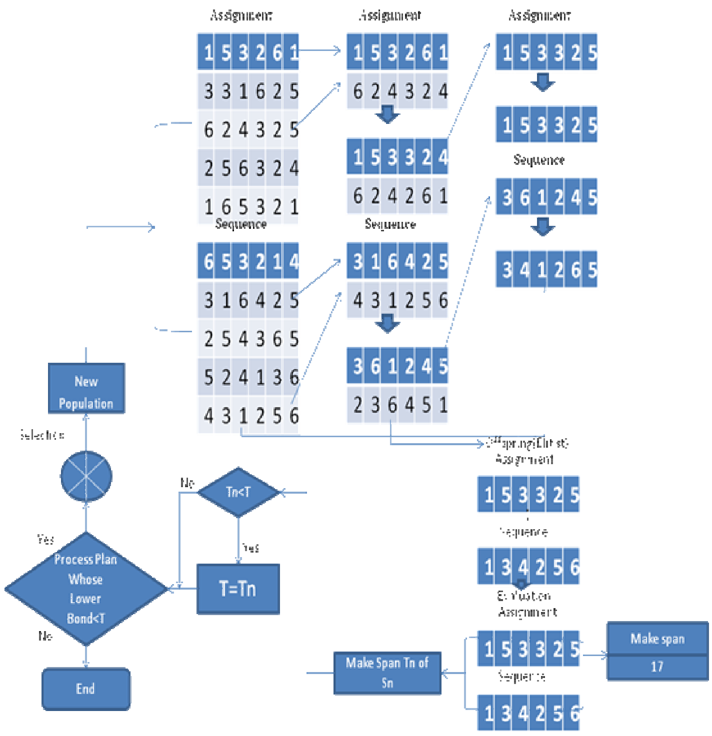


Fig. 2: Step wise model development for GA.

## 5. SCHEDULING PROBLEMS & RESULTS

### **5.1.** ***Scheduling problem 1 (partial flexibility)***

In this research the example of job shop scheduling problem data (partial flexibility) is taken [29].

Table1: Problem data for job shop, (partial flexibility)

| | Operations | Alternative Machines | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| Job 1 | $O_{11}$ | 2 | 3 | 4 | | - | - |
| | $O_{12}$ | - | 3 | - | 2 | 4 | - |
| | $O_{13}$ | 1 | 4 | 5 | - | - | - |
| Job 2 | $O_{21}$ | 3 | - | 5 | - | 2 | - |
| | $O_{22}$ | 4 | 3 | - | - | 6 | - |
| | $O_{23}$ | - | - | 4 | - | 7 | 11 |
| Job 3 | $O_{31}$ | 5 | 6 | - | - | - | - |
| | $O_{32}$ | - | 4 | - | 3 | 5 | - |
| | $O_{33}$ | - | - | 13 | - | 9 | 12 |
| Job 4 | $O_{41}$ | 9 | | 7 | 9 | - | - |
| | $O_{42}$ | - | 6 | - | 4 | - | 5 |
| | $O_{43}$ | 1 | - | 3 | - | - | 3 |

Table 2: Final chromosomes representations

| Number of cycle | Operation sequence | Machine selection |
|---|---|---|
| 100 | $O_{11}$ $O_{12}$ $O_{13}$ $O_{21}$ $O_{22}$ $O_{23}$ $O_{31}$ $O_{32}$ $O_{33}$ $O_{41}$ $O_{42}$ $O_{43}$ | 2 2 2 3 2 3 1 4 5 3 4 1 |
| 200 | $O_{11}$ $O_{12}$ $O_{13}$ $O_{21}$ $O_{22}$ $O_{23}$ $O_{31}$ $O_{32}$ $O_{33}$ $O_{41}$ $O_{42}$ $O_{43}$ | 2 2 2 5 2 6 1 4 5 3 4 1 |
| 800 | $O_{11}$ $O_{12}$ $O_{13}$ $O_{21}$ $O_{22}$ $O_{23}$ $O_{31}$ $O_{32}$ $O_{33}$ $O_{41}$ $O_{42}$ $O_{43}$ | 2 2 1 3 1 3 1 4 5 3 4 1 |
| 1000 | $O_{11}$ $O_{12}$ $O_{13}$ $O_{21}$ $O_{22}$ $O_{23}$ $O_{31}$ $O_{32}$ $O_{33}$ $O_{41}$ $O_{42}$ $O_{43}$ | 2 4 2 5 5 3 1 4 5 3 2 6 |
| 1500 | $O_{11}$ $O_{12}$ $O_{13}$ $O_{21}$ $O_{22}$ $O_{23}$ $O_{31}$ $O_{32}$ $O_{33}$ $O_{41}$ $O_{42}$ $O_{43}$ | 2 2 2 5 2 6 1 4 5 3 4 3 |

There are four jobs; each job has a three different operation, six different machines, and the alternative routing and processing time are given in table1. In this example, operation 1 of job 1 i.e. $o_{11}$ can be performed on machine 1, 2 or 3. The time taken by each machine is 2, 3 or 4 units of time, and so on.
The program was written in MATLAB using the genetic algorithms and selected the best makespan value for the job by using the number of cycle. Observer could see that the last job process on machine, final makespan value for this problem is 17. In these schedules of job on the different machine are different so that user get the different value of mean flow time, utilization of machine and average utilization of machine. Thus out of these schedule users select according to priority objective, schedule of job on machine.
Initially run the program on different number of cycle and selected out of the best suitable schedule. The final chromosomes are formed through this schedule is given below in table 2.The columns 1 through 12 represent the operation number of four jobs, three operations and the column 12 through 24 represent the corresponding machine associated through it. The makespan value plotted against the number of cycle as shown here and also compared with number of iterations. Gantt chart and makespan for the above scheduling problem is given (Fig. 2 to Fig. 11) for different number of iterations. The result of the problem is summarized in table 3.

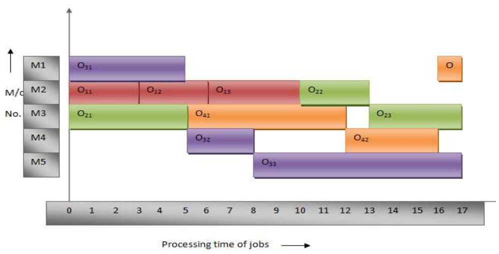


Fig. 3: Gantt chart for 100 no of cycle.

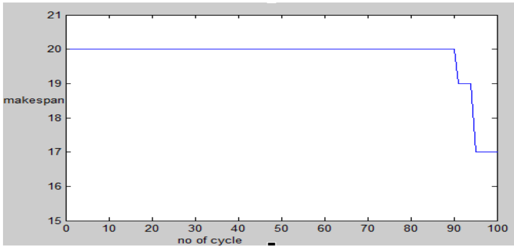


Fig. 4: Makespan vs. the 100 no of iterations.

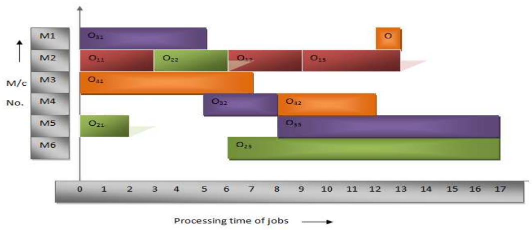


Fig. 5: Gantt chart for 200 no of cycle.

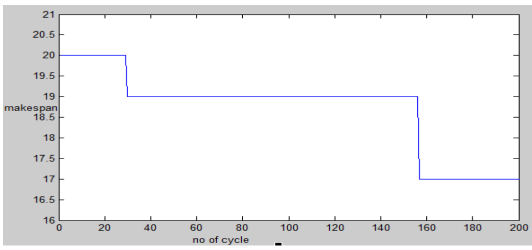


Fig. 6: Makespan vs. the 200 no of iterations.

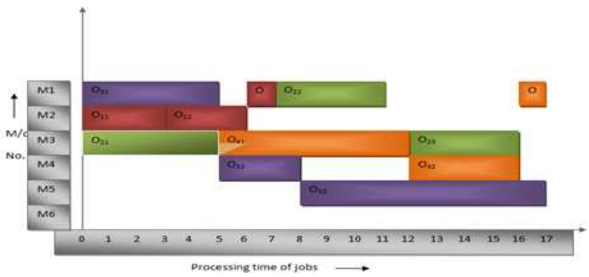


Fig. 7: Gantt chart for 800 no of cycle.

Table 3: Results of problem 1

| Number of cycle | Minimum makespan | Mean flow time | Utilization of machine | | | | | | Average utilization |
|---|---|---|---|---|---|---|---|---|---|
| | | | $M_1$ | $M_2$ | $M_3$ | $M_4$ | $M_5$ | $M_6$ | |
| 100 | 17 | 12.75 | 0.35 | 0.76 | 0.94 | 0.41 | 0.53 | 0.00 | 0.55 |
| 200 | 17 | 13.75 | 0.35 | 0.76 | 0.41 | 0.41 | 0.65 | 0.65 | 0.54 |
| 800 | 17 | 12.25 | 0.65 | 0.35 | 0.94 | 0.41 | 0.53 | 0.00 | 0.48 |
| 1000 | 17 | 13.50 | 0.29 | 0.76 | 0.65 | 0.29 | 1.00 | 0.18 | 0.53 |
| 1500 | 17 | 14.25 | 0.30 | 0.76 | 0.41 | 0.65 | 0.65 | 0.00 | 0.46 |

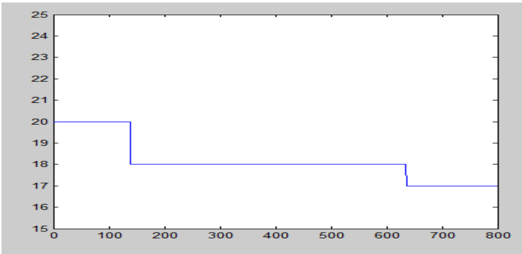

Fig 8: Makespan vs. the 800 no of iterations.

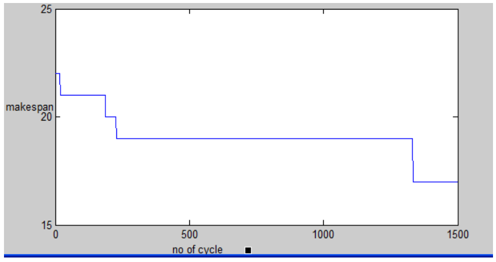


Fig. 12: Makespan vs. the 1500 no of iterations.

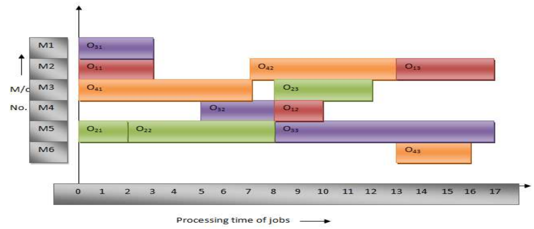


Fig. 9: Gantt chart for 1000 no of cycle.

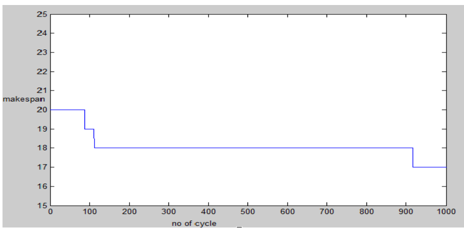

Fig. 10: Makespan vs. the 1000 no of iterations.

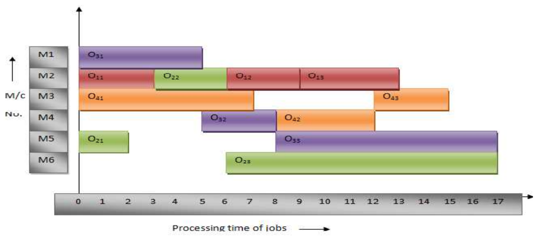


Fig 11: Gantt chart for 1500 no of cycle.

In the above result (Table 3.) for different number of cycle makespan value is minimum. The utilization of machine for each cycle is different so user can select minimum utilization of machine and corresponding to minimum cost.

### 5.2. *Scheduling problem 2 (total flexibility)*

In problem (2) data for job shops scheduling (total flexibility) are taken [30]. In this problem, initially taken two jobs and five machines are taken in to the consideration for testing purpose of job that schedule on machine. This problem is different from the previous problem. In this case all machines are ready to perform the operation. In the table 4 operation 1 of job 1 i.e. $o_{11}$ can be performed on any of five machine 1, 2, 3, 4 or 5, and so on.

Table 4: Problem data for job shop, (total flexibility)

| | Operations | Alternative Machines (total flexibility) | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| Job 1 | $O_{11}$ | 1 | 8 | 3 | 7 | 5 |
| | $O_{12}$ | 3 | 5 | 2 | 6 | 4 |
| | $O_{13}$ | 6 | 7 | 1 | 4 | 3 |
| job 2 | $O_{21}$ | 1 | 4 | 5 | 3 | 8 |
| | $O_{22}$ | 2 | 8 | 4 | 9 | 3 |

Table 6: Result of problem 2

| Number of cycle | Minimum makespan | Mean flow time | Utilization of machine | | | | | Average utilization |
|---|---|---|---|---|---|---|---|---|
| | | | $M_1$ | $M_2$ | $M_3$ | $M_4$ | $M_5$ | |
| 500 | 6 | 4.5 | 0.67 | 0.00 | 0.17 | 0.00 | 0.67 | 0.30 |

For solving of this problem initial generated chromosomes is equal to the number of total operation done by the machine. Here initially two jobs and five machines are taken in to the consideration. The job 1 has three operations and job 2 has two operations, so length of chromosomes is equal to the five. After 500 generation thee best chromosomes has been obtained.

Table 5: Best possible chromosomes

| Number of cycle | Operation sequence | Machine selection |
|---|---|---|
| 500 | $O_{11}O_{12}O_{13}O_{21}O_{22}$ | 1 5 3 1 1 |

The above chromosomes clearly shows the job 1 having three operation can be performed on the 1,5or 3 machine and the job 2 can be performed all the operation on machine 1.

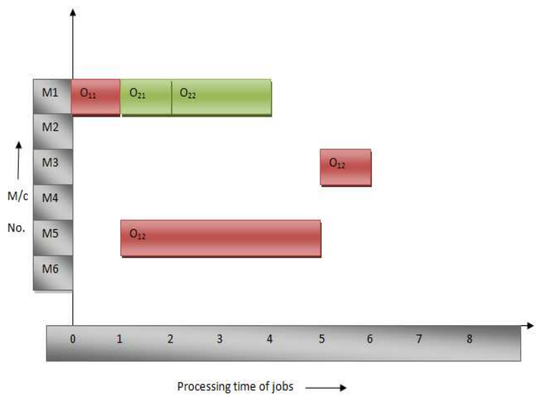


Fig. 13: Gantt chart schedule for 500 iterations.

From the above table make span of the problem is calculated makespan = 6. Gantt chart of the above scheduling is given above. Gantt chart is graphical representation of scheduling of jobs, In Gantt Chart time is placed at the abscissa and machine number is placed at the ordinate. The result of this problem is shown in Table 6.

## 6. CONCLUSIONS AND FUTURE WORK

This paper presents a MATLAB programming-based approach for the job shop (partial flexibility or total flexibility) scheduling problem by utilizing the various alternatives machines. The ability of a GA to provide multiple optimal solutions was exploited to generate a knowledge base of good solutions. Also genetic algorithms successfully schedule the best optimal solution of the job on the machine. This paper developed different permutation chromosomes for machine assignment and a sequence chromosome for order of job has been presented for alternative process plan for job shop scenario. The Performance analysis of this job shop scheduling approach is compare from the Literature. This objective has been achieved by competitive with other various methods. This has been efficient in the job shop scheduling problem addressed in [29].The schedule obtained have makespan value near to optimal. The other objectives also are obtained like mean flow time, utilization of machine, average utilization of machine etc. based on these objective users can give the priority of these objectives that which is required for us. The benefit of this method is that there exist at most three machines per operation that may replace each other, as the cost of these relative machines.

Further experiment in different kind of problem related to job shop scheduling and various genetic search strategies is needed. The numerical example showed are aim to minimize makespan time, with other several objectives has to be obtained. In this thesis some other several objective can be implemented. Also, the use of artificial techniques in different way also can be developed.

## REFERENCES

[1] Wilhelm W, Shin H, 1985. Effectiveness of alternative operations in a flexible manufacturing system [J]. International Journal of Production Research, 23(1): 65−79.

[2] Hankins S L, Wysk R A, Fox K R, 1984. Using a CATS database for alternative machine loading [J]. Journal of Manufacturing Systems, 3: 115−120.

[3] Nasr N, Elsayed E A, 1990. Job shop scheduling with alternative machines [J]. International Journal of Production Research, 28(9): 1595−1609.

[4] Liang, M. and Dutta, S., 1990. A mixed-integer-programming approach to the machine loading and process planning problem in a process layout environment. Int. J. Prod. Res., 28,1471–1484.

[5] Hutchison, J., Leong, K., Snyder, D. and Ward, P., 1991. Scheduling approaches for random job shop flexible manufacturing systems. Int. J. Prod. Res., 29, 1053–1067.

[6] Dauzère-Pérès S, Paulli J, 1997. An integrated approach for modeling and solving the general multiprocessor job-shop scheduling [J]. Annals of Operations Research, 70(3): 281−306.
[7] Kim K H, Egbelu P J, 1998. A mathematical model for job shop scheduling with multiple process plan consideration per job [J]. Production Planning & Control, 9: 250−259.
[8] Candido M A B, Khator S K, Barcia R M, 1998. A genetic algorithm based procedure for more realistic job shop scheduling problems [J]. International Journal of Production Research, 36(12): 3437−3457.
[9] Hussain M F, Joshi S B, 1998. A genetic algorithm for job shop scheduling problems with alternate routing [C]// IEEE International Conference on Systems, Man and Cybernetics. San Diego, CA, USA, 2225−2230.
[10] Jawahar N, Aravindan P, Ponnambalam S G, 1998. A genetic algorithm for scheduling flexible manufacturing systems [J]. International Journal of Advanced Manufacturing Technology, 14(8): 588−607.
[11] Kim K H, Egbelu P J, 1999. Scheduling in a production environment with multiple process plans per job [J]. International Journal of Production Research, 37(12): 2725−2753.
[12] H. Chen, J. Ihlow, C. Lehmann, 1999. A genetic algorithm for flexible job-shop scheduling. IEEE International conference on Robotics and Automation, Detroit, 2: 1120- 1125.
[13] Beck J C, Fox M S, 2000. Constraint-directed techniques for scheduling alternative activities [J]. Artificial Intelligence, 121(1/2):211−250.
[14] Thomalla C S, 2001. Job shop scheduling with alternative process plans [J]. International Journal of Production Economics, 74(1):125−134.
[15] Zhou, H., Feng, Y. and Han, L., 2001. The hybrid heuristic genetic algorithm for job shop scheduling. Comput. Ind. Engng, 40, 191–200.
[16] Choi I C, Choi D S, 2002. A local search algorithm for job shop scheduling problems with alternative operations and sequence dependent setups [J]. Computers & Industrial Engineering, 42(1): 43−58.
[17] I. Kacem, S. Hammadi, P. Borne, 2002. Pareto-optimality approach for flexible job-shop scheduling problems: Hybridization of evolutionary algorithms and fuzzy logic. Mathematics and Computers in Simulation, 60: 245-276.
[18] Kim Y K, Park K, Ko J, 2003. A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling [J]. Computers & Operations Research, 30(8): 1151−1171.
[19] Kim, Y., Kim, J. and Shin, K., 2004. The integration of FMS process planning and scheduling using an asymmetric multileveled symbiotic evolutionary algorithm. J. Kor. Inst. Ind. Engrs, 30, 130–145.
[20] Zhang, H., Gen, M, 2005;"Multistage-Based Genetic Algorithm for Flexible Job-Shop Scheduling Problem,"Journal of Complexity International, Vol. 11, pp. 223–232.
[21] Tavakkoli-Moghaddam R, Khodadadeghan Y, Haghnevis M, 2006. Efficient algorithm for a multi-criteria parallel machine scheduling problem with sequence-dependent setup times [C]// Proceedings of 5th International Symposium on Intelligent Manufacturing Systems. Sakarya, Turkey: Sakarya University, 159−167.
[22] Demir H I, 2006. Scheduling with alternative routes [C]// Proceedings of 5th International Symposium on Intelligent Manufacturing Systems. Sakarya, Turkey: Sakarya University, 1323−1338.
[23] Mehrabad M S, Fattahi P, 2007. Flexible job shop scheduling with tabu search algorithms [J]. International Journal of Advanced Manufacturing Technology, 32(5/6): 563−570.
[24] Zhang, W.C., Zheng, P.E., Wu, X.D, 2007; "Solution to Flexible Job Shop Scheduling Problems with Capacitated Constraints Based on ant Colony & Genetic Algorithms," Computer Integrated Manufacturing Systems, Vol. 13(2), pp. 333–337.
[25] F. Pezzella, G. Morganti, G. Ciaschetti, 2008. A genetic algorithm for the flexible job-shop scheduling problem. Computers & Operations Research, 35(10): 3202-3212.
[26] Xing, L.N., Chen, Y.W., Yang, K.W, 2009; "Multi-Objective Flexible Job Shop Schedule: Design and Evaluation by Simulation Modeling," Applied Soft Computing, Vol. 9, pp. 362–376.
[27] Xing, L.N., Chen, Y.W., Yang, K.W, 2009; "An Efficient Search Method for Multi Objective Flexible Job Shop Scheduling Problems," Journal of Intelligent Manufacturing, Vol. 20, pp. 283–293.
[28] K.-H. Kim And P. J. Egbelu, 2010. Scheduling in a production environment with multiple process plans per Job, International Journal of Production Research, 37:12, 2725-2753.
[29] Chaudhry, I. A., 2012. Job shop scheduling problem with alternative machines using genetic algorithms. Journal of Central South University, 19(5), 1322-1333.
[30] Mesghouni, K.,Hammadi, S., & Borne, P. 2004. Evolutionary algorithms for job-shop scheduling. International Journal of Applied Mathematics and Computer Science, 14(1), 91-104.