Genetic Algorithm for Job Shop Scheduling

Mr.P.P.Bhosale
Department Of Computer Science and Engineering,
SVERI's College Of Engineering Pandharpur, Solapur University Solapur

Mr.Y.R.Kalshetty
Department Of Computer Science and Engineering,
SVERI's College Of Engineering Pandharpur, Solapur University Solapur

Abstract – The study of Job shop scheduling to find optimal solution is presented in this paper. The main objective of this paper is to minimize the total completion time that is makespan of the entire job is to minimize. To find optimal solution many methods are designed, a new genetic algorithm is designed to find optimal solution for job shop scheduling problem. Initial population is generated randomly and chromosomes are created. The genetic operators (crossover, mutation) were used and the code was written in MATLAB to achieve the objective of this paper that is minimized makespan. The input parameters were sequence of operation of each job on machine and processing time of each operation are provided. The Proposed approach is tested on several standard problem instances and result was compared with other approaches, the results shows effectiveness of proposed algorithm.
Key Words: Scheduling, initial population, crossover, mutation.

I. INTRODUCTION

In any manufacturing system scheduling is an important activity. The main purpose of scheduling is to allocate resources to the operations, where set of n job must be processed on set of m machine. The job shop scheduling is the allocation of machines to process the collection of jobs to minimize the makespan. The makespan is the total time required to complete all the jobs. The optimal solution of JSSP is very difficult to find so it is considered as NP-hard problem. There are many methods were used to solve the scheduling problem which includes Differential evolution[1], Tabu search[2], simulated annealing[3], Ant colony optimization, Tabu search, shifting bottleneck[4], genetic algorithm[8][9], neural network, particle swarn optimization etc. The objective of this paper is to minimize total completion time that in minimize makespan. In this paper we propose genetic algorithm to achieve the objective. Some researchers have already studied the job shop scheduling by using genetic algorithm. Genetic algorithm was firstly introduced by De jong and Holland which is based on natural selection mechanism. A hybrid GA (HGA) mechanism is used by Linet O zdamar[15] in which HGA is integrated with traditional scheduling tools for project scheduling problems.

The main purpose of this paper is to find a schedule that is, an allocation of operation to time intervals to machine which having minimum length. Here we propose Genetic algorithm to find the optimal solution. The major genetic operator is local search algorithm which start from initial solution called as population and applies genetic operators on it to find more optimal solution than previous. The process repeated until it reaches to maximum iteration limit and each new chromosome corresponds to a solution.

II. LITRATURE REVIEW

Eui-Seok Byeon[5] investigate heuristic method which update the parameters by using lower bound and upper bound for the corresponding schedule. Nguyen, Mengjie Zhang et al.[7] proposed genetic algorithm to automatically find new dispatching rules for the job shop problem. PN controllers methodology is designed for forbidden state problem including uncontrollable transition and liveness requirement by Vittaldas V. Prabhu[9]. Regions theory is used to design PN controller. Hiroyasu Toba[10] solved problem of redundant blocking time by using WIP estimation flow control method. Haoxun Chen, Chengbin Chu [11] used efficient pseudo-polynomial time dynamic programming algorithm to relaxed job level sub problems, Peter B. Luh et al. [12] used combining recurrent neural network idea including Lagrangian relaxation for constraint handling to propose novel Lagrangian relaxation neural network for optimization problem. S. Hajri, N. Liouane, S. Hammadi, and P. Borne [13] proposed a controlled genetic algorithm (CGA) based on fuzzy logic to solve job-shop scheduling problems. To achieve best solution heuristic rules for generating initial solution and new methodology was used. Emma Hart and Peter Ross[14] used genetic algorithm to find optimal solution for job shop problem and explained how crossover and mutation operations produce final result.

III. METHODOLOGY

A. *Problem Statement:*

Instance of 3 X 3 job shop scheduling is given below

3 3
0 2 1 1 2 1
0 1 2 0 1 3
1 3 2 2

Example: Instance of 3 jobs and 3 machines.

Jobs	Machine no and Processing Time		
J0	**M0(3)**	**M1(2)**	M2(2)
J1	**M0(2)**	**M2(1)**	M1(4)
J2	**M1(4)**	**M2(3)**	

Table - 1 Sequence of jobs on machine with processing time

Job shop scheduling contains of finite set of n jobs consists of chain of operation and finite set of m machines, each machine can handle at most one operation at a time. Each operation needs to be processed during an uninterrupted period of a given length on given machine. Job J {J1, J2, and J3…….Jn} is the set of independent job. A job Ji contain different operations Oi1, Oi2, Oi3…..Oim to be processed one after another in the given sequence. The set of machines M= {M1, M2 …Mm} is given to process these operations. The jobs and machines are available at time 0 and only one operation can be performed on each machine at a time. The objective of this problem is to minimize makespan.

Genetic algorithm is local search algorithm starts from initial solution called as population and applies genetic operators like Chromosome representation, crossover and mutation on it to find more optimal solution than previous. The process repeated until it reaches to maximum iteration limit and each new chromosome corresponds to a solution. Following are the different steps to evaluate genetic algorithm

IV. IMPLEMENTATION

A. A Genetic Algorithm for Job Shop Scheduling

Genetic algorithm is local search algorithm starts from initial solution called as population and applies genetic operators on it to find more optimal solution than previous. The process repeated until it reaches to maximum iteration limit and each new chromosome corresponds to a solution. Following are the different steps to evaluate genetic algorithm.

a. Generation of initial population

To find optimal solution for job shop problem initial population plays vital role. Initial population has been produced randomly in any genetic algorithm and after that best individual is selected and old one is replaced with new generated population

b. Chromosome Representation

The chromosomes representation is very important. It will be number, array of number, matrix or any other data structure. The chromosomes of the individuals of the initial population are randomly generated using the symbols representing the functions and terminals chosen to solve the particular problem. By doing this we find different initial combinations.one of the combination is as follows.

1	**2**	**1**	**2**	**1**	**0**	**0**	**0**

Table 2 Chromosome representation for 3 x 3 instances

A chromosome is given as [12121000] for 3 job and 3 machines problem. Here, 0 indicates operation of job J0, 1 indicates operation of job J1, and 2 indicate operation of job J2. The number 0 and 1 appears three times and 2 appear 2 times in chromosome because there are three operations in 0^{th} and 1^{st} job and 2^{nd} job contain 2 operations. The number 1 is repeated the three times in a chromosome; it shows three operations of job J1. The first number 1 indicates that first operation of job J1. This processes on the machine 1. The second number 1 shows the second operation of job J1 which processes on the machine 2, and so on.

This chromosome can be represented as follows.

1	2	1	2	1	0	0	0
1,0	2,0	1,1	2,1	1,2	0,0	0,1	0,2

Table - 3 Chromosome representation for 3 x 3 instances

The first row represents job numbers and second row represents two numbers one is job number and other indicates machine number.

c. Crossover

Genetic algorithm produces Chromosome which having good schedule. Crossover operator is required to find more optimal solution in GA. Two random (parent) chromosomes are mixed together and produce new chromosome to produce more feasible solution than previous one by getting best characteristics

d. Mutation

Mutation is performed by interchanging the position of two sub operations from the selected chromosome without disturbing the precedence constraints.

B. Flow Chart of Genetic Algorithm for JSP

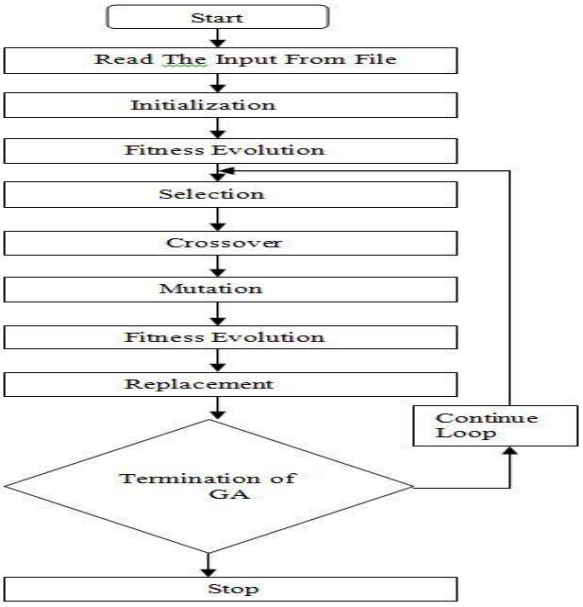

Figure 1. Flow Chart of Genetic Algorithm for JSP

V. RESULT AND DISCUSSION

The code was generated in matlab using proposed Genetic algorithm and tested on following test instance [16]

Jobs	Machine no and Processing Time		
J1	M1(3)	M2(2)	M3(2)
J2	M1(2)	M3(1)	M2(4)
J3	M2(4)	M3(3)	

Table 4 Machine no with Processing Time

The following output was obtained at the end of 100th iteration

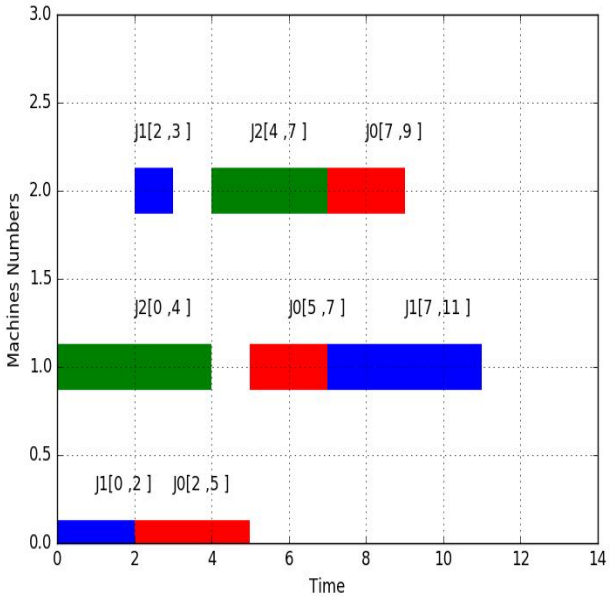

Figure 2 Result of 3 X 3 instances.

The optimized schedule of obtained minimum makespan is as follows,

1	**2**	**1**	**0**	**2**	**0**	**0**	**1**

Table - 5 Chromosome representation for 3 x 3 instance

The chromosome given above can be represented with respect to (i, j).

1	**2**	**1**	**0**	**2**	**0**	**0**	**1**
1,0	**2,1**	**1,2**	**0,0**	**2,2**	**0,1**	**0,2**	**1,1**

Table 6 Chromosome representation for 3 x 3 instances

A. Computational Result

Proposed genetic algorithm is compared with some benchmark instances up to 500 iterations. The table 7 shows optimal solutions of Genetic Algorithm.

Dataset	N*M	Benchmark	GA	GES	TS	SA
La16	**10*10**	**945**	**1078**			
La17	**10*10**	**784**	**861**			
La18	**10*10**	**848**	**954**			
La19	**10*10**	**842**	**1057**	**842**	**842**	**842**
La20	**10*10**	**902**		**902**	**902**	**902**
Orb1	**10*10**	**1059**	**1390**			
Orb2	**10*10**	**888**	**1083**			
Orb3	**10*10**	**1005**	**1542**			
Orb4	**10*10**	**1005**	**1243**			
Orb5	**10*10**	**887**	**1500**			
Orb6	**10*10**	**1010**	**1420**	**1010**	**1010**	**1010**
Orb7	**10*10**	**397**	**523**	**397**	**397**	**397**
Orb8	**10*10**	**899**	**1351**	**899**	**899**	**899**
Orb9	**10*10**	**934**	**1157**	**934**	**934**	**934**
Orb10	**10*10**	**944**	**1502**	**944**	**944**	**944**

Abz5	**10*10**	**1167**	**1392**			
Abz6	**10*10**	**925**	**1086**			

Table - 7 Computational results with benchmark instances.

VI. CONCLUSION

The proposed Genetic Algorithm is used to solve job shop scheduling problem and to find optimal solution chromosome structure is developed and crossover, mutation operators are used. Gantt chart is used to produce the result. Proposed algorithm is examined with different benchmark problem instance and the computational result shows that proposed GA can obtain near about solution and shows effectiveness of this approach.

REFERENCES

[1] Antonin Ponsich, Carlos A. Coello,"A hybrid Differential Evolution—Tabu Search algorithm for the solution of Job-Shop Scheduling Problems," Applied Soft Computing 13 (2013) 462-474.

[2] S.G.Ponnambalam, P. Aravindan and S.V.Rajesh "A Tabu Search Algorithm for Job Shop Scheduling"Int J Adv Manuf Technol (2000) 16: 765 – 771.

[3] M.emin aydin and Terenc C. Fogaty, "A simulated annealing algorithm for multi-agent systems:Ajob shop scheduling application," Journal of Intelligent Manufacturing, (2004), 15: 805-814.

[4] Wen-Qi Huang and Zhi Huang, "Algorithm Based on Taboo Search and Shifting Bottleneck for Job Shop Scheduling," J Comput . Sci. & Technol.(2014) 19:776 – 781.

[5] Eui-Seok Byeon, S. David Wu, and Robert H. Storer, "Decomposition Heuristics for Robust JSP," IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 14, NO. 2, APRIL 1998

[6] Wen-Qi Huang and Zhi Huang, "Algorithm Based on Taboo Search and Shifting Bottleneck for Job Shop Scheduling," J Comput . Sci. & Technol.(2014) 19:776 – 781vghghcgh

[7] Su Nguyen, Mengjie Zhang, Mark Johnston et al., "A Computational Study of Representations in Genetic programming to evolve dispatch rules for JSSP," IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 17, NO. 5, OCTOBER 2013

[8] Noorul Haq, K.Balasubramanian, B. Shashidharan, and R.B. Karthica K. Oka, and Y. Tagawa, "Parallel line job shop scheduling using genetic algorithm," Int J Adv Manuf Technol (2008) 35: 1047 – 1052.

[9] Vittaldas V. Prabhu., "Stable Fault Adaptation in Distributed Control of Hetrachical Manufacturing JSP" IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 19, NO. 1, FEBRUARY 2003

[10] Hiroyasu Toba, "A Tight Flow Control for Job-Shop Fabrication Lines with finite buffers," IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING , VOL. 2, NO. 1, JANUARY 2005

[11] Haoxun Chen, Chengbin Chu, and Jean-Marie Proth., "An Improvement of the Lagrangean Relaxation Approach for JSS:A dynamic method," IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 14, NO. 5, OCTOBER 1998

[12] Peter B. Luh, Xing Zhao,Yajun Wang, and Lakshman S. Thakur, "Lagrangian Relaxation Neural Networks for JSS," IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 16, NO. 1, FEBRUARY 2000

[13] S. Hajri, N. Liouane, S. Hammadi, and P. Borne, "A Controlled Genetic Algorithm by Fuzzy Logic and Belief functions for JSS," IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 30, NO. 5, OCTOBER 2000

[14] Emma Hart and Peter Ross ,"GAVEL—A New Tool for Genetic Algorithm visualization," IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 5, NO. 4, AUGUST 2001.

[15] Linet O zdamar . "A Genetic Algorithm Approach to a General Category Project Scheduling Problem" IEEE TRANSACTIONS ON SYSTEMS MAN AND CYBERNETICS, MARCH 1999.

[16] "jobshop1.", http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/jobshop1.txt.