Vol. 5(14) Jan. 2015, PP. 2056-2061

Contents list available at IJMEC

International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)

Journal Homepage: www.aeuso.org

PISSN: 2411-6173 - EISSN: 2305-0543

Solving open shop scheduling problem using Genetic Algorithm

Alireza Ghoonodi

Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran

Corresponding Author's Email: ghonodi@iausari.ac.ir

Abstract

This paper considers the scheduling problem of open shop to minimize makespan. Purpose of scheduling open shop problem is attaining to a suitable order of processing n jobs by m specified machines such that makespan be minimized. Open shop scheduling problem has very large and complex solution space and so is one of NP-Problems. Up till now, different algorithms have been presented for open shop scheduling problem. In this paper we use Genetic Algorithm (GA) as a strategy for solving the scheduling open shop problem and the *results show* better effectiveness in some instances of proposed algorithm compared with other algorithm.

Keywords: *Open Shop, Genetics Algorithm, Scheduling, NP-Complete*

1. Introduction

In open shop scheduling problem, each job j, j=1,2,…,n, with release time $r_{j,}$ which is the earliest time when the job is available and in an on-line environment also the time when the job is known in the system, has to be processed on m machines once. The order in which job j passes through the m machines is unimportant. The processing of job j on machine I, i=1,2,…,m, is denoted as operation O(i,j) with a processing time P(i,j). it is assumed that the processing time are bounded by P_{max} and i.i.d. (independently and identically distributed) random variables. At any given time each machine can handle at most one job and each job can be processed on at most one machine. Preemption is forbidden, that is, any commenced operation has to be completed without interruption. And no job is delayed during the load time of each machine. The completion time of job j on machine I is denoted by C(i,j). the objective is to find a sequence of jobs with the given processing times on each machine to minimize the makespan, i.e., the maximum completion time[1]. Many researchers have presented different algorithms which we point out some of them as follows:

O(n) for two machines[2] A linear time algorithm to obtain a minimum finish time schedule for the two-processor open shop together with polynomial algorithm to obtain a minimum finish time preemptive schedule for open shops with more than two processors are obtained. Pinedo [3] offered another simple distribution rule known as "Longest Alternate Processing Time (LAPT) "، which solves problem for two machines in polynomial time. Brucker et al. [4] extended other branch and connected algorithm to general problem of m machine. Fiala presented polynomial time algorithm which solves problem for m arbitrary machines[5]. Among innovative algorithms, Alcaide and his coworkers [6] presented a tab searching algorithm for minimizing makespan in scheduling open shop problem. Liaw[7] developed a connected genetics algorithm for solving scheduling open shop

Article History:

Received Date: Dec. 03, 2014
Accepted Date: Mar. 19, 2015
Available Online: Mar. 30, 2015

problem in order to minimizing makespan . Prins[8] achieved a fine solution for solving scheduling open shop problem by providing a genetics algorithm. Barzegar et al. [9] used combined genetics algorithm as a strategy for solving scheduling open shop problem and compared proposed algorithm with DGA algorithm. Results showed that the proposed algorithm has better effectiveness than DGA algorithm. Ahmadizar and Farahani[10] , developed a hybrid GA for the open shop scheduling problem with the objective of minimizing the makespan. An open shop scheduling problem with sequence-dependent setup times was examined by Noori-Darvish and Tavakkoli-Moghaddam [11]. A novel bi-objective mathematical programming was designed in order to minimize the total tardiness and the makespan.

Zobolas et al. [12] developed a hybrid genetic algorithm that solves the open shop scheduling problem to minimize the makespan. Matta [13] studied proportionate multi-processor open shop (MPOS) scheduling problem to minimize the makespan. They developed two different mixed integer linear programming formulations for proportionate MPOS to minimize makespan. While the first formulation is time-based model, second formulation is sequence-based model. Then, they proposed Genetic Algorithm (GA) for proportionate MPOS and finally presented its computational performance analysis. Goldansaz et al. [14] consider multi-processor open shop scheduling problems with independent setup time and sequence dependent removal time.

Among above mentioned methods, genetics algorithm has been considered as one of the well-known applied algorithms. In this paper, a new algorithm is presented as combination of genetics algorithm with tab searching algorithm. This combination causes better searching is performed in problem solution space in order to achieving better scheduling. The second section has problem explanation has presented. Then, in third section proposed algorithm has discussed and in forth section, implementation results have been offered and finally conclusion is provided.

2. Problem explanation

Scheduling open shop problem includes n jobs and m machines and each job includes m operations which should be implemented in its corresponded machine in already determined time. In other words, first operation of job j should be implemented by first machine and second operation of job j should be implemented by second machine and so on. Purpose of scheduling open shop problem is attaining a suitable combination of job processing, such that makespan be minimized [15,16, 17]. Table 1 shows typical system of scheduling open shop problem which includes three jobs and each job includes 2 operations which should be implemented.

Table 1: A typical system of open shop

Jobs	Machine 1	Machine 2
Job 1	4	3
Job 2	5	1
Job 3	2	6

For example, here, operation 2 form job 2 which its required running time is 1, should be implemented by machine 2. Figure1 illustrates Gant diagram of table 1 typical system.

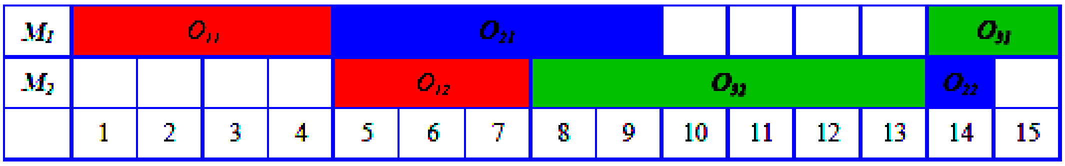

Figure1: Gant Diagram, Sample system of Table1

3. Proposed Algorithm

In proposed method is used of combination of genetics algorithm with local searching algorithm as a strategy for solving scheduling open shop problem which improves searching in state-space and its objective is minimizing makespan. Figure 2 shows total stages of purposed algorithm.

3.1 Chromosome demonstration

In proposed algorithm, it has been used one dimensional array in the length of makespan for chromosome is an unique integer number which is obtained by numbering job operations. Figure 3 shows a typical chromosome for typical system of Table 1, as you could see in structure of chromosome of figure 3, each gene presents operation number that should be processed by regarded machine. In this way, each chromosome shows one scheduling for makespan.

1	2	3	4	5	6
O_{11}	O_{21}	O_{12}	O_{32}	O_{31}	O_{22}

Figure 3: Structure of chromosome

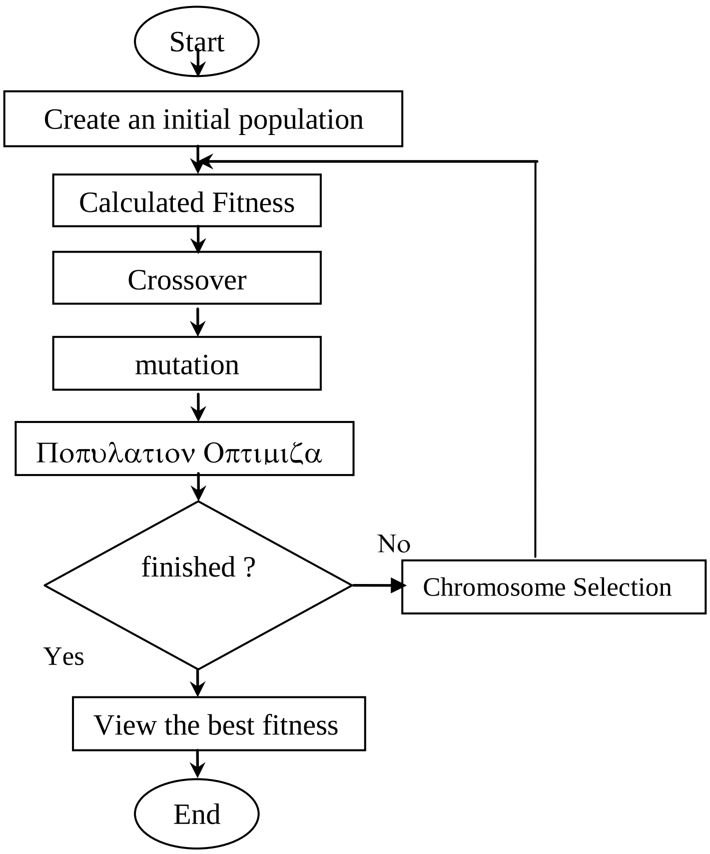

Figure2: Overall the proposed algorithm

3.2 Fitness

Chromosome fitness is considered as required time for makespan. In proposed algorithm we used follows formulation for calculating fitness.

$$\text{Fitness} = \text{Max}_{1<i<=N} \{ T_i \}$$

In this formula, N is number of jobs and T is makespan.

3.3 Parents Selection

In proposed algorithm, it has been used ordering method for selecting parents. In this technique, all chromosomes are ordering based on fitness. This means that each chromosome have less C_{Max} selected and deleted chromosomes. The selection procedure for the next generation of chromosomes in the hybrid algorithm is proposed. In selecting the combination of the chromosomes are arranged based on fitness and remove duplicate chromosome deserve better than 20% of randomly selected for the next generation population.

3.4 Crossover-Over Operator

In proposed algorithm, after selecting 2 parents for crossover operator, we select a random number in 1 to n interval (n is number of operators) and those genes which are in range of 0 to random number shifts from primary parent and similar genes of second parent are eliminated and reminded genes are inserted in empty houses ordinary in figure 4.

3.5 Mutation Operator

After selecting parent chromosome from existent population, 2 randomized points in interval 1 to n (n is number of operators) are selected and available genes in the range of selected points one shifted to the left rotary. Typical mutation has been presented in figure5.

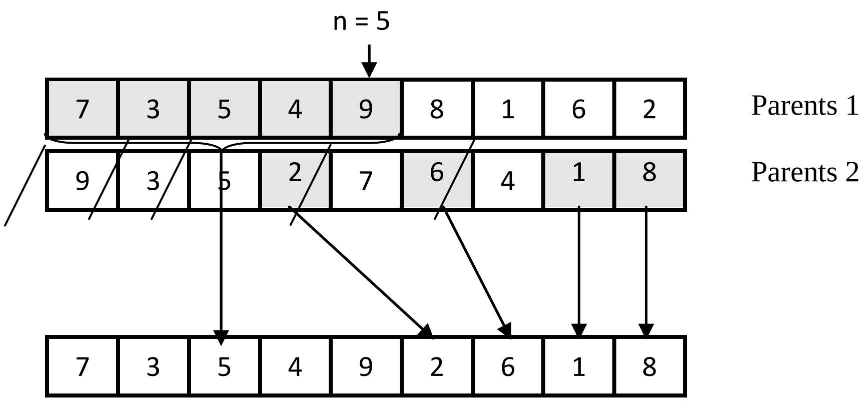

Figure 4: Crossover operator for typical system [9]

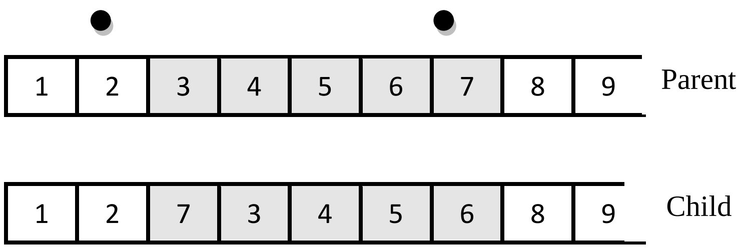

Figure 5: Applying mutation operator for sample chromosome [9]

3.6 Chromosome selection

Chromosome selection for next generation in proposed algorithm is combinative. In selecting based on combinative method, first , chromosomes one ordered based on fitness and repeated chromosomes are eliminated and then 30% of those chromosomes with better fitness are selected and reminded chromosomes are selected randomly for next generation of population.

3.7 Stop constraint

In proposed algorithm constraint of stopping algorithm is limiting number of generation means that if number of generation reaches to regard number, algorithm steps. Implementation Result: For implementation proposed algorithm it has been used C#.Net 2008 programming language. Algorithm applied on an Intel® Pentium ® processor 2.4 GHz and RAM 4.00 GB. Four series of test data have been defined for measuring effectiveness of proposed algorithm. The detailed results of proposed algorithm on the benchmark problems from Taillard [18] are presented in Table 2. These problems are all small and medium and high sized problems.

Table 2: The results proposed algorithm benchmark Taillard [18]

Instance	High	Proposed algorithm
*Taillard 4*4-1*	*193*	*193*
*Taillard 4*4-2*	*236*	*236*
*Taillard 4*4-3*	*271*	*271*
*Taillard 4*4-4*	*250*	*250*
*Taillard 4*4-5*	*295*	*294*
*Taillard 4*4-6*	*189*	*187*
*Taillard 4*4-7*	*201*	*201*
*Taillard 4*4-8*	*217*	*218*
*Taillard 4*4-9*	*261*	*263*
*Taillard 4*4-10*	*217*	*218*
*taillard 5*5*1*	*300*	*301*
*taillard 5*5*2*	*262*	*263*
*taillard 5*5*3*	*328*	*239*
*taillard 5*5*4*	*310*	*315*
*taillard 5*5*5*	*329*	*330*
*taillard 5*5*6*	*312*	*314*
*taillard 5*5*7*	*305*	*307*
*taillard 5*5*8*	*300*	*302*
*taillard 5*5*9*	*353*	*350*
*taillard 5*5*10*	*326*	*327*
*taillard 7*7*1*	*438*	*340*
*taillard 7*7*2*	*449*	*456*
*taillard 7*7*3*	*479*	*485*
*taillard 7*7*4*	*467*	*470*
*taillard 7*7*5*	*419*	*418*
*taillard 7*7*6*	*460*	*462*
*taillard 7*7*7*	*435*	*437*
*taillard 7*7*8*	*426*	*430*
*taillard 7*7*9*	*460*	*462*

taillard 7*7*10	*400*	*401*
taillard 10*10-1	*637*	*635*
taillard 10*10-2	*588*	*588*
taillard 10*10-3	*598*	*598*
taillard 10*10-4	*577*	*570*
taillard 10*10-5	*640*	*641*
taillard 10*10-6	*538*	*539*
taillard 10*10-7	*616*	*617*
taillard 10*10-8	*695*	*670*
taillard 10*10-9	*595*	*593*
taillard 10*10-10	*596*	*590*

Conclusion

In this paper, a new method has been presented based on genetics algorithm for solving scheduling open shop problem. The proposed algorithm causes that searching performs in better space. Also, using different genetics operators of proposed algorithm causes that diversity of chromosomes is maintained always and effectiveness in finding better solutions increased. Experimental results show that proposed algorithm has better effectiveness in some instances.

References

[1] Low, C., & Yeh, Y. (2009). Genetic algorithm-based heuristics for an open shop scheduling problem with setup, processing, and removal times separated.*Robotics and Computer-Integrated Manufacturing*, *25*(2), 314-322.
[2] Gonzalez, T., & Sahni, S. (1976).Open shop scheduling to minimize finish time. *Journal of the ACM (JACM)*, *23*(4), 665-679.
[3] Pinedo, M. Scheduling: Theory, Algorithms and Systems, prentice-Hall, Englewood Cliffs,NJ, 1995.
[4] Brucker, P., Hurink, J., Jurisch, B., & Wöstmann, B. (1997). A branch & bound algorithm for the open-shop problem. *Discrete Applied Mathematics*, *76*(1), 43-59.
[5] Sevast'janov, S. V. E. (1994). On some geometric methods in scheduling theory: a survey. *Discrete Applied Mathematics*, *55*(1), 59-82.
[6] Alcaide, D., Sicilia, J., & Vigo, D. (1997).A tabu search algorithm for the open shop problem. *Top*, *5*(2), 283-296.
[7] Liaw, C. F. (2000).A hybrid genetic algorithm for the open shop scheduling problem. *European Journal of Operational Research*, *124*(1), 28-42.
[8] Prins, C. (2000).Competitive genetic algorithms for the open-shop scheduling problem. *Mathematical methods of operations research*, *52*(3), 389-411.
[9] Barzegar, B., Motameni, H., & Divsalar, A. (2012). A Hybrid Genetic Algorithm for the Open Shop Scheduling with Makespan and Total Completion Time.*Journal of Advances in Computer Research*, *3*(4), 17-24.
[10] Ahmadizar, F., & Farahani, M. H. (2012).A novel hybrid genetic algorithm for the open shop scheduling problem. *The International Journal of Advanced Manufacturing Technology*, *62*(5-8), 775-787.
[11] Noori-Darvish, S., & Tavakkoli-Moghaddam, R. (2012).Minimizing the total tardiness and makespan in an open shop scheduling problem with sequence-dependent setup times. *Journal of Industrial Engineering International*, *8*(1), 1-13.
[12] Zobolas, G. I., Tarantilis, C. D., & Ioannou, G. (2009).Solving the open shop scheduling problem via a hybrid genetic-variable neighborhood search algorithm. *Cybernetics and Systems: An International Journal*, *40*(4), 259-285.
[13] Matta, M. E. (2009).A genetic algorithm for the proportionate multiprocessor open shop.*Computers & Operations Research*, *36*(9), 2601-2618.
[14] Goldansaz, S. M., Jolai, F., & Anaraki, A. H. Z. (2013).A hybrid imperialist competitive algorithm for minimizing makespan in a multi-processor open shop.*Applied Mathematical Modelling*, *37*(23), 9603-9616.
[15] Low, C., & Yeh, Y. (2009). Genetic algorithm-based heuristics for an open shop scheduling problem with setup, processing, and removal times separated.*Robotics and Computer-Integrated Manufacturing*, *25*(2), 314-322.
[16] Bräsel, H., & Hennes, H. (2004).On the open-shop problem with preemption and minimizing the average completion time. *European Journal of Operational Research*, *157*(3), 607-619.
[17] Lin, H. T., Lee, H. T., & Pan, W. J. (2008). Heuristics for scheduling in a no-wait open shop with movable dedicated machines. *International Journal of production economics*, *111*(2), 368-377.
[18] Taillard, E. D. (1994). Parallel taboo search techniques for the job shop scheduling problem. *ORSA journal on Computing*, *6*(2), 108-117.