International Journal of Science and Engineering Applications
Volume 10-Issue 02, 17 - 20, 2021, ISSN:- 2319 - 7560

Extended Hybrid Genetic Algorithm for Solving Job Shop Scheduling Problem

Sumaia E. Eshim
Benghazi University, Engineering Faculty,
Industrial and Manufacturing System Engineering Department
Benghazi, Libya

Mohammed M.Hamed
Benghazi University, Engineering Faculty,
Industrial and Manufacturing System Engineering Department
Benghazi, Libya

Abstract: In this paper, a hybrid genetic algorithm (HGA) to solve the job shop scheduling problem (JSSP) to minimize the makespan is presented. In the HGA, heuristic rules are integrated with genetic algorithm (GA) to improve the solution quality. The purpose of this research is to investigate from the convergence of a hybrid algorithm in achieving a good solution for new benchmark problems with different sizes. The results are compared with other approaches. Computational results show that a hybrid algorithm is capable to achieve good solution for different size problems.

Keywords: Extended a hybrid genetic algorithm; Job Shop Scheduling; Makespan

1. INTRODUCTION

Scheduling is defined as the allocation of resources over time to perform a collection of tasks. Most scheduling problems are complex combinatorial optimization problems and they are mainly very difficult to solve. Job shop scheduling problem (JSSP) is a well know one of the hardest combinatorial optimization problems where the main goal is find a schedule with minimized makespan for processing of n jobs on a set m machines. Since the problem is well known as NP-Hard class and therefore no deterministic algorithms can solve them in a reasonable amount of time [1]. That means, exact algorithms, such as (branch and bound method, dynamic programming) can be used only for small size problems. Therefore, more complex problem must be solved by heuristic methods. Successful heuristic methods include approaches based on simulated annealing [2], tabu search [3], and genetic algorithms [4 ,5]. These approaches have been employed to deal with complex scheduling problems which are capable of producing high quality solutions with a reasonable computational effort [4].

Genetic algorithms were first proposed by Holland in the 1970s and have been successfully used in a variety of problems. Genetic algorithm (GA) is a heuristic search that mimics the process of natural evolution. The first applied a genetic algorithm to the JSSP in 1985 successfully, and now genetic algorithms have been proved to be an effective approach for the JSSP. First applied a genetic algorithm to the JSSP in 1985 successfully, and now genetic algorithms have been proved to be an effective approach for the JSSP as mention by Janes et al [6]. The genetic algorithm is an effective meta-heuristic method to solve combinatorial optimization problems, thus many researchers have applied them also to the scheduling problems [7 - 10]. The implementation time of the GA can be defined as the time required by the algorithm to render an optimal or satisfactory solution. This time reflects the solution quality comprising each generation. If the quality of the solutions is poor, i.e. the individuals are beyond the fitness function or imposed constraints, then the results seem to be hopeful, but, the GA will take more time to render or reach the best solution. Some techniques and operators are used to improve the solution quality. The performance of GAs solutions depends on the quality of the initial population [7, 11] on which the quality and performance of the next populations generations will depend on. All previous algorithms and approaches, there are positives and negatives of every optimization method. Therefore, combinations of two or more techniques are used to solve JSSP. These methods are called hybrid algorithms. Hybrid methods are frequently employed for solving JSSP for example, hybrid GA and heuristic rules [14], hybrid GA and local search [9], and GA with simulated annulling [12,13]. The hybrid algorithms perform better than its corresponding individual counterparts as with the hybridization convergence rate is usually high and it also helps in escaping local minima. All the algorithms developed for JSSP have their strengths and weaknesses [14]. Therefore, researchers are constantly in search of new algorithms and a lot of efforts have been put in to optimize and improve existing methods. Our intention in this article is to investigate the effects of the varying size of problem on hybrid genetic algorithm proposed by (Boushaala, et. al., 2013) [15] for solving JSSP. To check the effectiveness of a hybrid genetic algorithm, the results obtained is compared with other approaches.

The remaining contents of this research article are organized in different sections as follows. Section 2 covers the conventional problem definition. Section 3 discusses a hybrid genetic algorithm. Section 4 provides experimental results and analysis. Finally, the conclusion is presented in section 5.

2. PROBLEM DEFINITION

The job shop scheduling problem (JSSP) can be described as follows: a set of n jobs (J) to be scheduled on a set of m machines (M). Each job j visits a number of machines in a predetermined order. The processing times for each job at each machine are given and no machine can process more than one job at a time. If a job is started on a machine, then it cannot be interrupted. The problem is finding a schedule of the jobs on the machines. The assumptions of the present problem are:

- Every job has a unique sequence on *m* machines. There are no alternate routings;
- There is only one machine of each type in the shop;
- Processing times for all jobs are known and constant;
- All jobs are available for processing at time zero;
- Machine absences are not allowed;
- Transportation time between machines is zero;
- Each machine can perform only one job at a time;
- Each job visits each machine only once;
- An operation of a job can be performed by only one machine;
- Operation cannot be interrupted;
- A job does not visit the same machine twice;
- An operation of a job cannot be performed until its preceding operations are completed;
- There is no restriction on queue length for any machine;
- There are no limiting resources other than machines/workstations;
- The machines are not identical and perform different operations. [15].

The objective of the scheduling job is to optimize a certain criterion. This criterion is used as a performance measure of the schedule. A common objective criterion is to minimize the makespan (C_{max}), which is the time needed to complete all the jobs.

3. A HYBRID GENETIC ALGORITHM

Genetic algorithm (GA) for job shop scheduling is an optimization method of searching based on evolutionary process which works with a population of solutions. In this paper, a hybrid genetic algorithm (HGA) based on an integration of a genetic algorithm and some developed and recommended heuristic rules is used. Also, the effects of the implemented new benchmark problem with different size on the proposed algorithm are investigated to check the effectiveness of a hybrid algorithm against the problem size variety. This algorithm is described in detail in paper [15]. The flowchart of a hybrid Genetic algorithm is shown in Figure 1.

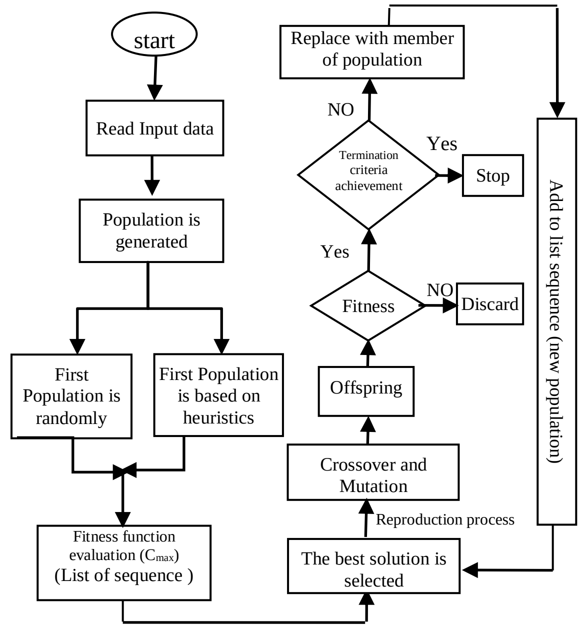

Fig:1 Proposed algorithm flow chart

4. COMPUTATIONAL EXPERIMENTS

This section gives a detailed explanation about the date used and the results obtained. For this study, the benchmark problems with different sizes were considered to investigate the efficiency of a hybrid genetic algorithm (HGA) against the problem size variety. These problems have varying sizes with the number of jobs varying from 4 to 6 and the number of machines varying from 3 to 10. There are 25 benchmark problems, 5 each of sizes 4×3, 5×4, 5×6, 6×6, and 5×10. The HGA is implemented to obtain the best solutions. The results of HGA are compared with Recommended Heuristic Rules (RHR), Developed Heuristic Rules (DHR), and GA based on randomization Algorithm (GA-R) addressed in the paper [15], The result comparison is presented in table 1. In this table the first column presents number of problem. The second column presents makespan from RHR while the third column presents makespan from DHR. The fourth column presents makespan from GA-R while the fifth column presents makespan from HGA.

Table 1. Result Comparison of Different Approaches

Problem number	RHR	DHR	GA-R	HGA
4×3				
1	23	26	23	23
2	26	26	26	26
3	21	21	21	21
4	21	21	21	21
5	22	22	22	22
5×4				
1	24	29	23	24
2	25	25	26	25
3	29	33	29	28
4	21	21	21	21
5	26	26	29	26
5×6				
1	29	29	29	27
2	33	34	33	31
3	39	39	39	39
4	30	33	33	27
5	31	31	30	29
6×6				
1	37	36	38	36
2	31	33	36	30
3	37	39	37	34
4	29	29	29	29
5	27	27	27	27
5×10				
1	52	53	55	51
2	44	44	44	42
3	42	43	43	41
4	33	33	32	32
5	30	32	31	26

From the above table it is evident that the performance of the HGA is better than other approaches. As it can be seen, the HGA could obtain the best solutions in most benchmark problems and compete with compared algorithms. It achieves the best for 24 instances of all the thirty considered instances (96%), compared to 44% when applying GA based on randomization population from the domain search space. In case of applying only the developed heuristic rules the best solution is obtained by 44%, while it is 44% when applying the recommended heuristic rules.

Figs. 2–6 show the results of benchmarks 4×3, 5×4, 5×6, 6×6, and 5×10 obtained using the HGA and other approaches, respectively. The comparison between HGA and other approaches for all benchmark problems are presented in Fig.7. As evident from these figures, the HGA could solve the problem considered for this study efficiently and compete with compared approaches.

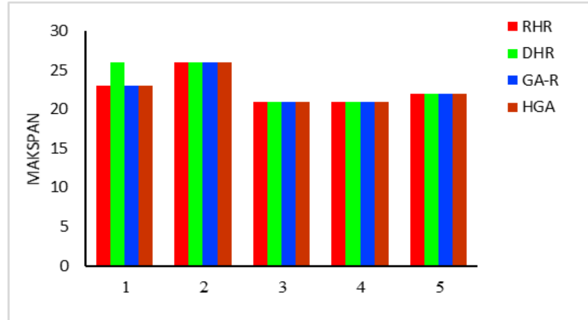

Figure 2. 4×3 problem

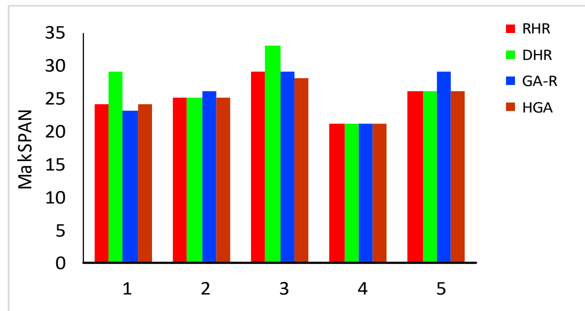

Figure 3. 5×4 problem

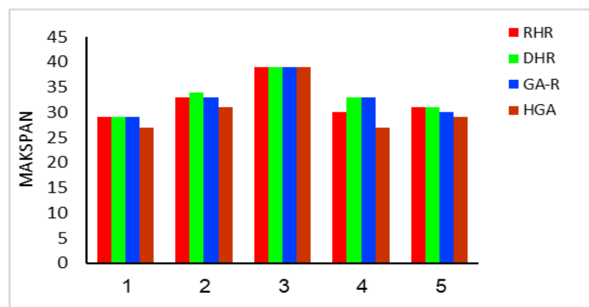

Figure 4. 5×6 problem

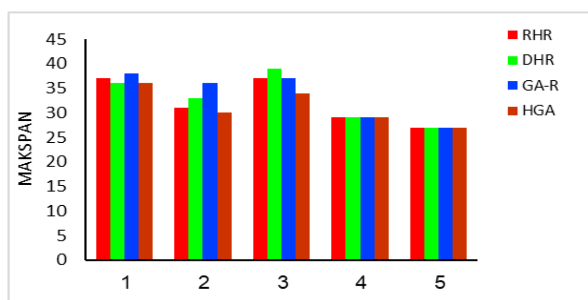

Figure 5. 6×6 problem

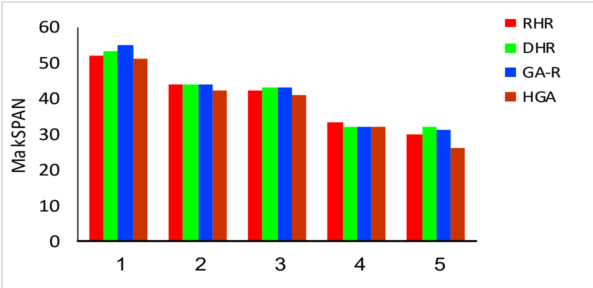

Figure 6. 5×10 problem

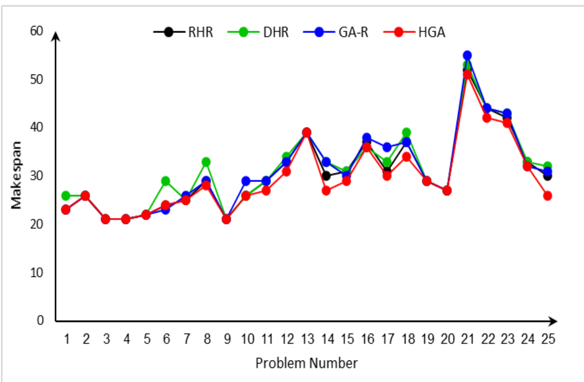

Figure 7. The comparison results obtained in Table 1

5. CONCLUSION

In this paper, the effects of selected new benchmark problem with different sizes on a hybrid genetic algorithm (HGA) for solving job shop scheduling problem (JSSP) to minimize the makespan are investigated**.** Also, the HGA is compared with the other approaches, and the results showed the HGA could find the best solutions for all kinds of problems and could find them in shorter computational times compared to the other approaches. In other words, there is a gradual growth in the makespan and it is evidently seen when the machine size starts increasing. When the machine size is less as 3 machines, all the approaches behave similarly in producing the results. But when the machine size increases, the HGA algorithm gives the best result compared to other approaches.

6. REFERENCES

[1] Kuczapski , M., Micea1 , V., Maniu , A., Cretu , A, 2010. "Efficient Generation of Near Optimal Initial Population to Enhance Genetic Algorithm for Job-Shop Scheduling", ISSN 1392 – 124X information technology and control.

[2] Yazdani . M , Gholami . M , Zandieh . M , and Mousakhani . A , 2009. " Simulated Annealing A algorithm for Flexible Job-Shop Scheduling Problem " Journal of applied sciences 9(4):662-670.

[3] Nowicki, E., and Smutnicki, C., 2005. " An Advanced Tabu Search Algorithm for the Job Shop Problem ", Journal of Scheduling, Vol. 8, pp. 145-159.

[4] Norozi, A., Ariffin, N., and Ismai l., N., 2010. "Application of Intelligence Based Genetic Algorithm for Job Sequencing Problem on Assembly Line ", ISSN 1941-7020, American J. of Engineering and Applied Sciences 3 (1): 15-24.

[5] Rafinejad, S., Ramtin , F., Arabani , A., 2009."A New Approach to Generate Rules in Genetic Algorithm Solution to a Job Shop Schedule by Fuzzy Clustering", Proceedings of the World Congress on Engineering and Computer Science Vol. II, San Francisco, USA.

[6] Janes, G., Perinic,M., Jurkovic, Z., 2017. "Applying Improved Genetic Algorithm For Solving Job Shop Scheduling Problems" , ISSN 1330-3651 (Print), ISSN 1848-6339. Tehnički vjesnik X 1243-1247.

[7] Kemppainen, K., 2005."Priority Scheduling Revisited Dominant Rules, Open Protocols, and Integrated Order Management", Helsinki School of Economics.

[8] Tamilarasi, A., kumar, T., 2010. "An Enhanced Genetic Algorithm with Simulated Annealing for Job-Shop Scheduling", International Journal of Engineering, Science and Technology Vol. 2, No.1, pp.144-151.

[9] Kalshetty, R., Y., Adamuthe, R., Y., and Kumar, S., P, 2020. "Genetic Algorithms with Feasible Operators for Solving Job Shop Scheduling Problem", Journal of Scientific Research, Volume 64, Issue 1, India.

[10] Hongjing, W., Li, S., Jiang, H., Hu, J., Hu, J., 2018. "Hybrid Genetic Simulated Annealing Algorithm for Improved Flow Shop Scheduling with Makespan Criterion", Appl. Sci. 8, 2621; doi:10.3390/app8122621.

[11] Lin, S., Goodman, E., Punch, W., , 2003. "A Genetic Algorithm Approach to Dynamic Job Shop Scheduling Problems", Proceedings of the Seventh International Conference on Genetic Algorithms, PP. 481−488.

[12] Tamilarasi, A., kumar, T, 2010. "An Enhanced Genetic Algorithm with Simulated Annealing for Job-Shop Scheduling", International Journal of Engineering, Science and Technology Vol. 2, No.1, pp.144-151.

[13] Hongjing, W., Li, S., Jiang, H., Hu, J., Hu, J., 2018. "Hybrid Genetic Simulated Annealing Algorithm for Improved Flow Shop Scheduling with Makespan Criterion", Appl. Sci. 8, 2621; doi:10.3390/app8122621.

[14] Qiu, X., Lau, H., 2014."An AIS-based hybrid algorithm for static job shop scheduling problem" J. Intell. Manuf. 25 (3), 489–503.

[15] Boushaala, A. A., Shouman, M. A., and Esheem , S., 2012. "Genetic Algorithm based on Some Heuristic Rules for Job Shop Scheduling Problem" Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management Istanbul, Turkey.