

A new approach to solving multiobjective flowshop scheduling problems: A multi-MOORA-based genetic algorithm

Alparslan Serhat Demir* and Mine Büşra Gelen**

* Department of Industrial Engineering, Faculty of Engineering, Sakarya University, Sakarya, TURKEY

** Program of Industrial Engineering, Institute of Natural Sciences, Sakarya University, Sakarya, TURKEY

* Corresponding Author : alparslanserhat@sakarya.edu.tr

Submitted : 10/06/2019

Revised : 22/04/2021

Accepted : 09/05/2021

ABSTRACT

Flowshop scheduling problems constitute a type of problem that is frequently discussed in the literature, where a wide variety of methods are developed for their solution. Although the problem used to be set as a single purpose, it became necessary to expect more than one objective to be evaluated together with increasing customer expectation and competition, after which studies started to be carried out under the title of multiobjective flowshop scheduling. With the increase in the number of workbenches and jobs, the difficulty level of the problem increases in a nonlinear way, and the solution becomes more difficult. This study proposes a new hybrid algorithm by combining genetic algorithms, which are metaheuristic methods, and the Multi-MOORA method, which is a multicriterion decision-making method, for the solution of multiobjective flowshop scheduling problems. The study evaluates and tries to optimize the performance criteria of maximum completion time, average flow time, maximum late finishing, average tardiness, and the number of late (tardy) jobs. The proposed algorithm is compared to the standard multiobjective genetic algorithm (MOGA), and the Multi-MOORA-based genetic algorithm (MBGA) shows better results.

Keywords: Multiobjective decision-making; Flow-shop scheduling; Multiobjective genetic algorithm; Multi-MOORA.

INTRODUCTION

A flowshop scheduling problem is a type of scheduling problem that is frequently discussed in the field of production planning and control. n jobs as $job_1, job_2, job_3, \dots, job_n$ are processed in the predetermined order in the machine. The benches can only do one job at one time. The processing time of each job denoted as i , on each bench denoted as j , is shown as t_{ij} , and each job must be completed before the predetermined delivery date d_i . Naturally, as the number of jobs increases, the problem becomes more difficult to solve, and finding the optimal solution takes a long time. For this reason, the problem is called NP-Hard (Ponnambalam et al., 2004).

Although initial studies on flowshop scheduling problems were usually carried on one objective, criteria such as makespan C_{max} , average flowtime \bar{F} , and maximum tardiness T_{max} have been added to the objective function. Thus, the problem has become a multiobjective flowshop scheduling problem. In addition to the large number of

benches in multiobjective flowshop scheduling problems, it is difficult to solve the problem because of the excessively large number of objectives that are to be optimized together. This study proposed a new hybrid algorithm by integrating the Multi-MOORA method, which is a multicriterion decision-making method for solving such problems into the genetic algorithm. The results of the proposed method were compared to those of the genetic algorithm, and it was found that the proposed method provides better results. C_{max} minimization of maximum completion time as objective function criteria, \bar{F} minimization of average flow time, minimization of maximum late finishing T_{max} , minimization of average tardiness \bar{T} , and maximization of the number of jobs that finished on time NTJ are discussed.

LITERATURE REVIEW

Studies on flowshop scheduling problems were initiated by Johnson (1954) with the consideration of two- and three-machine flowshop scheduling problems. The problem was generally dealt with as a single objective until the late 1990s. In these studies, the total flow time was aimed to be minimized mainly by the minimum makespan criteria [Rajendran et al., 1990; Chen et al., 1995; Murata et al., 1996; Das et al., 1995]. After this date, a variety of intuitive techniques have been used to solve flowshop scheduling problems, which are handled extensively. Among these studies, Ishibuchi and Murata (1996) proposed a multiobjective genetic local search algorithm and used it to optimize the criteria of makespan, maximum tardiness, and total flowtime. Pasupathy et al. (2006) minimized the makespan and total flowtime of jobs by using PGA-ALS, a multiobjective genetic algorithm that provides a local search based on pareto sequences. Pan, Wang and Qian (2009) described a hybrid algorithm based on differential evolution, and the effectiveness of the proposed algorithm was tested on the problem, where it was aimed to optimize the makespan and maximum tardiness criteria. Optimization of total the flowtime and makespan criteria was aimed by Keskin (2010), and a multiobjective hybrid genetic algorithm was developed for the solution of a no-wait flowshop scheduling problem. With the combination of a hybrid discrete differential evolution (HDDE) algorithm and variable neighborhood search (VNS), Mokhtari et al. (2011) minimized the criteria of maximum completion time and total cost of resources. A novel genetic algorithm (NGA) proposed by Pour, Tavakkoli-Moghaddam and Asadi (2013) was used to optimize the makespan, total waiting time, and total tardiness criteria. Marichelvam, Prabakaran and Yang (2014) minimized the weighted sum of makespan and mean flowtime criteria by the firefly algorithm. Li & Ma (2016) proposed a new multiobjective memetic search algorithm to solve the multiobjective permutation flowshop scheduling problem with makespan and total flowtime. Deng & Wang (2017) also proposed a competitive memetic algorithm to solve the multiobjective distributed permutation flowshop scheduling problem with makespan and total tardiness criteria. Robert & Rajkumar (2019) proposed a hybrid algorithm that includes genetic and simulated annealing algorithms to solve the biobjective permutation flowshop scheduling probe. Geng, Ye & Liu (2020) proposed a multiobjective memetic algorithm by setting the parameters with the Taguchi method in their study, where they focused on the multiobjective hybrid flowshop scheduling problem with dual resource constraints. They compared the results with NSGA-II, MODE, and MOMVO algorithms. Chachan & Hasanali (2021) solved the multiobjective permutation flowshop scheduling problem by proposing a hybrid algorithm that they use together with particle swarm and variable neighbourhood search algorithm to minimize cumulative completion time and total flow time. When previous studies are examined, it is seen that the genetic algorithm is preferred more in heuristics. There are only a few studies on the use of multiobjective decision-making techniques and genetic algorithms. Braglia and Grassi (2009) applied the TOPSIS method and the NEH (Nawaz-Enscore-Ham) heuristics in optimizing the makespan and maximum tardiness criteria, and the performance of the implemented model was tested by comparing it to the multiobjective genetic local search (MOGLS) algorithm. Lin, Lee and Wu (2012) provided a solution to a reentrant flowshop scheduling problem, which is aimed at optimization of the average and maximum deviation of makespan and order on-time rate criteria by AHP and the genetic algorithm. This study used the Multi-MOORA and genetic algorithm techniques together for solution of multiobjective flowshop scheduling problems.

MULTIOBJECTIVE GENETIC ALGORITHM

Genetic algorithms were first proposed by Holland in 1975, and this method attracted intensive interest of researchers with its emergence (Poon and Carter, 1995). It was inspired by the theory of evolution that embraced Darwin's idea of sustaining the existence of the fittest in nature (Gonçalves et al., 2005). Genetic algorithms generally consist of selection, crossover, mutation, and replacement process steps. The algorithm first begins with the creation of the initial population encoded as chromosomes. Individuals are then selected from the population to be crossed over in the direction of a particular election strategy. Crossed and mutated procedures are applied to the selected individuals to try to obtain better fitting individuals (solutions). The crossed-over and mutated individuals are replaced by some of the weak individuals in the initial population using a predetermined replacement strategy. The population size of genetic algorithms and selection, crossover, mutation, and replacement strategies affect the quality of the results obtained.

Although genetic algorithms have been used to solve a wide range of problems that are aimed to optimize a single objective, they have begun to be considered as multiobjective problem solution methods with the development of the industry and the need for assessment of multiple objectives together. One of the major studies on this subject is the study by Schaffer in 1984, who proposed a multiobjective Vector-Evaluated GA (VEGA) (Horn et al., 1994). When the literature in the following years is examined, it is seen that a few researchers tried to use the genetic algorithm for multiobjective problems (Ponnambalam et al., 2004). All the criteria considered in the multiobjective genetic algorithm are combined in a single function called the multiobjective function (Veeraiah et al., 2017). The following are the multiobjective functions needed to solve 5 objectives in a single function in the genetic algorithm, which are used to make comparisons to the proposed algorithm:

$$f(x) = \text{Min}[\alpha C_{max} + \beta \bar{F} + \gamma T_{max} + \delta \bar{T} + \theta NLJ] \quad (1)$$

$$\alpha \geq 0, \beta \geq 0, \gamma \geq 0, \delta \geq 0, \theta \geq 0$$

$$\alpha + \beta + \gamma + \delta + \theta = 1$$

C_{max} = The maximum completion time (makespan)

\bar{F} = Avarage flowtime

T_{max} = Maximum tardiness

\bar{T} = Avarage tardiness

NLJ = Number of late (tardy) jobs

$\alpha, \beta, \gamma, \delta, \theta$ Criteria weights

MULTI-MOORA METHOD

The MOORA plus the full multiplicative form (Multi-MOORA) method, which is a multicriterion decision-making method, is based on the multiobjective optimization on the basis of ratio analysis (MOORA) method, which was contributed to the literature by Brauers and Zavadskas (2006). The MOORA method was applied as a method of evaluating the results of the Ratio Method and the Reference Point Approach Method together. Brauers and Zavadskas (2010) developed the Multi-MOORA method by adding the Full Multiplicative Form to the MOORA method. The Multi-MOORA method, like Equation (2), requires an initial matrix, in which alternatives are evaluated based on various criteria. In the generated matrix, the criterion that is cost effective is specified as the minimum, and the benefit criterion is specified as the maximum criterion.

$$X = \begin{matrix} & C_1 & C_2 & \dots & C_n \\ & \text{Maximum or Minimum} & & & \\ A_1 & \left[\begin{matrix} x_{11} & x_{12} & \dots & x_{1n} \end{matrix} \right. \\ A_2 & \left. \begin{matrix} x_{21} & x_{22} & \dots & x_{2n} \end{matrix} \right. \\ \dots & \left[\begin{matrix} \dots & \dots & \dots & \dots \end{matrix} \right. \\ A_m & \left. \begin{matrix} x_{m1} & x_{m2} & \dots & x_{mn} \end{matrix} \right] \end{matrix} \tag{2}$$

The results obtained with the Ratio Method, the Reference Point Approach, and the Full Multiplicative Form are also evaluated with the Theory of Dominance proposed by the same authors to obtain the Multi-MOORA ranking. When a single order is obtained with the Theory of Dominance, the definitions of absolute dominance, general dominance, transitivity, overall being dominated, equability, and circular reasoning are utilized (Brauers and Zavadskas, 2012).

PROPOSED HYBRID MODEL: MULTI-MOORA-BASED GENETIC ALGORITHM

In order to benefit from the success of the proposed model in evaluating the quality of the solution, the Multi-MOORA method is used by hybridizing with the genetic algorithm. From the first step of the proposed hybrid algorithm, an initial population of number N is created. In the second step, for each individual in the population, the fitness value is calculated by using Equation (1). Then, m individuals with the best fitness value are selected for evaluation with the Multi-MOORA method. As in Equation (3), the selected m individuals are evaluated with the Multi-MOORA method from the perspectives of maximum completion time C_{max} (Minimization), average flowtime \bar{F} (Minimization), maximum late finishing T_{max} (Minimization), average tardiness \bar{T} (Minimization), and number of jobs finished on time NTJ (Maximization).

$$X = \begin{matrix} & C_{max} & \bar{F} & T_{max} & \bar{T} & NTJ \\ & Min & Min & Min & Min & Max \\ Individual_1 & \left[\begin{matrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \end{matrix} \right. \\ Individual_2 & \left. \begin{matrix} x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \end{matrix} \right. \\ \dots & \left[\begin{matrix} \dots & \dots & \dots & \dots & \dots \end{matrix} \right. \\ Individual_m & \left. \begin{matrix} x_{m1} & x_{m2} & x_{m3} & x_{m4} & x_{m5} \end{matrix} \right] \end{matrix} \tag{3}$$

During this evaluation, the ranking obtained by applying the Ratio Method, the Reference Point Approach, and the Full-Multiplicative Form is evaluated with the theory of dominance, and the ultimate Multi-MOORA ranking is achieved. The first two individuals in the Multi-MOORA rank are selected to be sent to the crossover stage. By repeating this process by the amount of crossings, the individuals that are needed for crossover are selected. The crossover and mutation procedures are then performed to obtain new individuals. The population size N is preserved by eliminating the increasing population based on a suitable value. The flowchart of the proposed hybrid algorithm is given in Fig. 1.

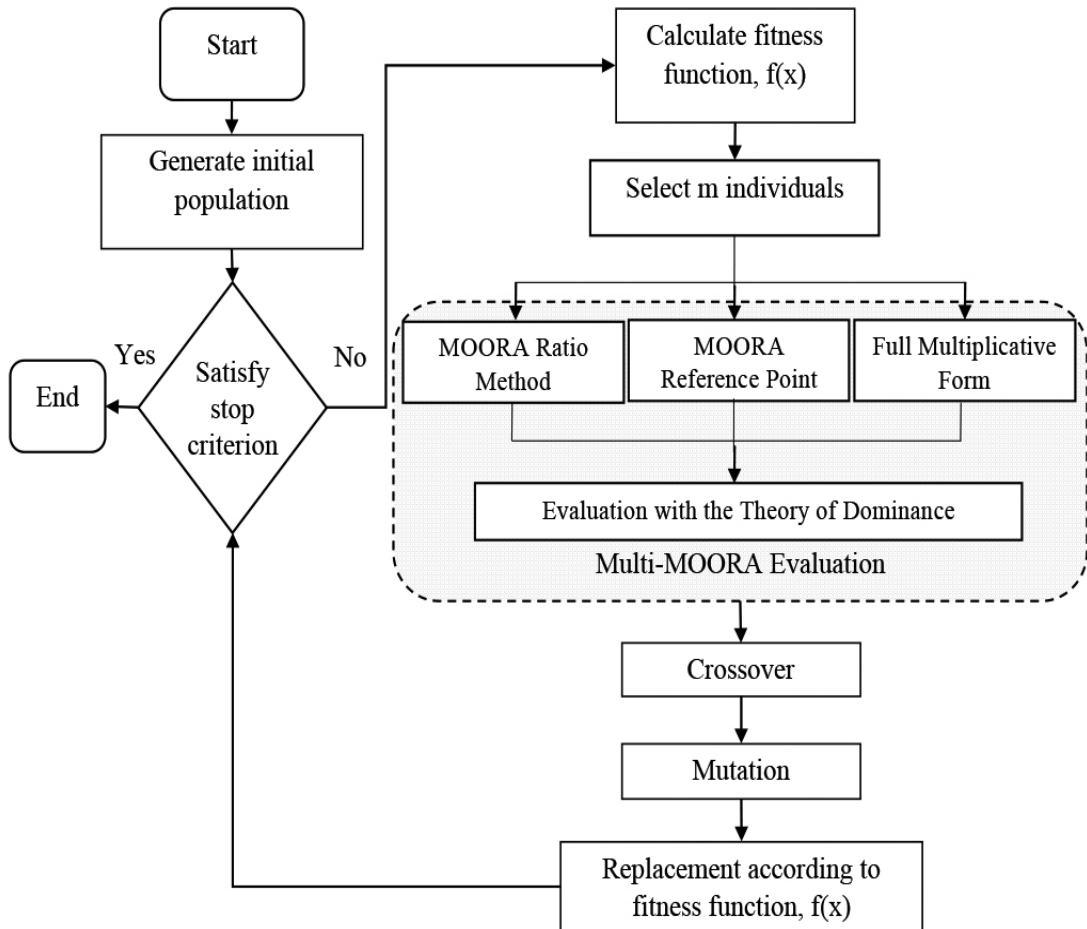


Figure 1: The Multi-MOORA-Based Genetic Algorithm (MBGA)

EXPERIMENTAL STUDY

In the implementation, a macro was developed in Microsoft Excel to compare the performances of the Multi-MOORA-Based Genetic Algorithm and the classical multiobjective genetic algorithm. The algorithms were tested on three different flowshop scheduling problems consisting of 25 jobs, 50 jobs, and 100 jobs. C_{max} , \bar{F} , T_{max} , \bar{T} and NLJ were used as the objective criteria, and the weight of each criterion was taken equal. In the crossover and mutation parts of the algorithms, a linear order crossover crossing strategy and an arbitrary three-job change mutation strategy were used, respectively. Population size was taken as $N = 40$, and the number of individuals to be evaluated at the Multi-MOORA stage was $m = 5$. In the MOGA method, the tournament width was selected as 5 in the selection strategy. The results obtained by running 50 generations of algorithms are given in Table 1 in detail.

Table 1: Comparison of the MBGA and MOGA methods in terms of different problem sizes

Problem Size	Methods	$f(x)_{min}$	C_{max}	\bar{F}	NLJ	\bar{T}	T_{max}
25	MBGA	222.832	616	374.48	11	24.68	88
	MOGA	230.344	604	377.36	13	33.36	124
50	MBGA	458.912	970	559.82	31	183.74	550
	MOGA	470.644	973	581.7	36	200.52	562
100	MBGA	951.724	1769	972.42	83	586.2	1348
	MOGA	968.606	1794	978.33	83	593.7	1394

When the results are examined, it is seen that the minimum fitness function $f(x)_{min}$ value is provided with MBGA for three different problem sizes. As the size of the problem grows, the superiority of the proposed MBGA method over the classical MOGA is significantly increased. In addition to the recommended MBGA method, $f(x)_{min}$, the effect of the average fitness value on each generation was investigated by $f(x)_{avg}$, and the results for three different problem sizes are shown in Fig. 2 to 4.

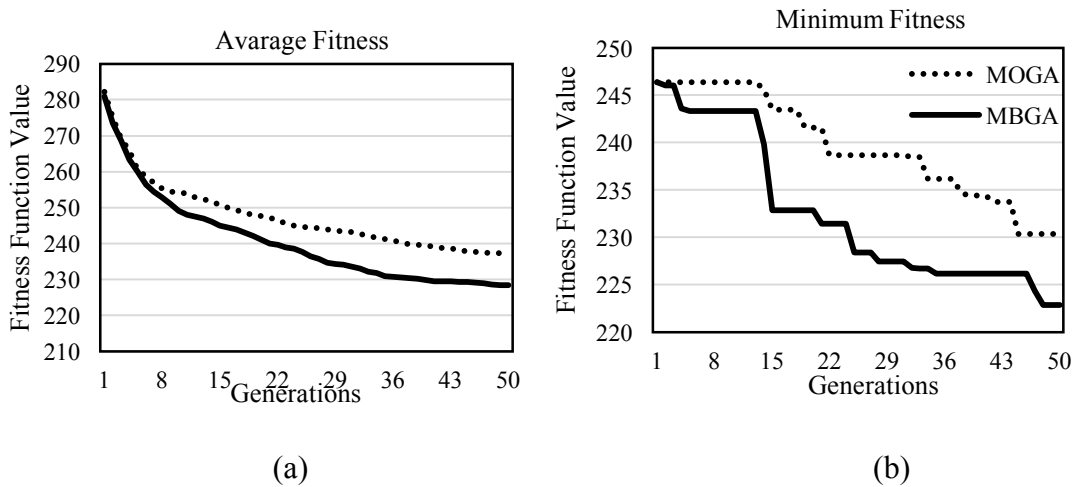


Figure 2: MBGA and MOGA for 25 jobs, $f(x)_{avg}$ comparison (a) $f(x)_{min}$ comparison (b)

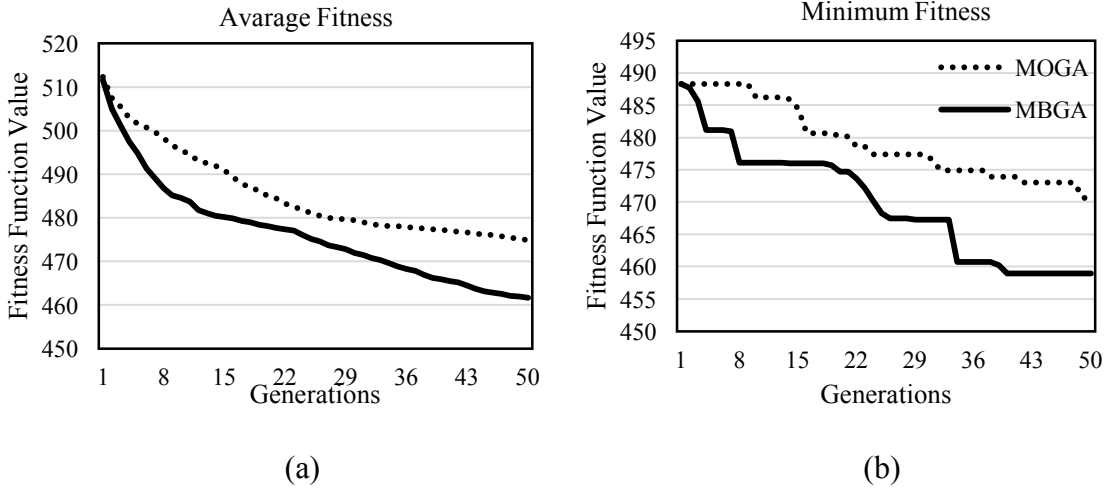


Figure 3. MBGA and MOGA for 50 jobs, $f(x)_{avg}$ comparison (a) $f(x)_{min}$ comparison (b).

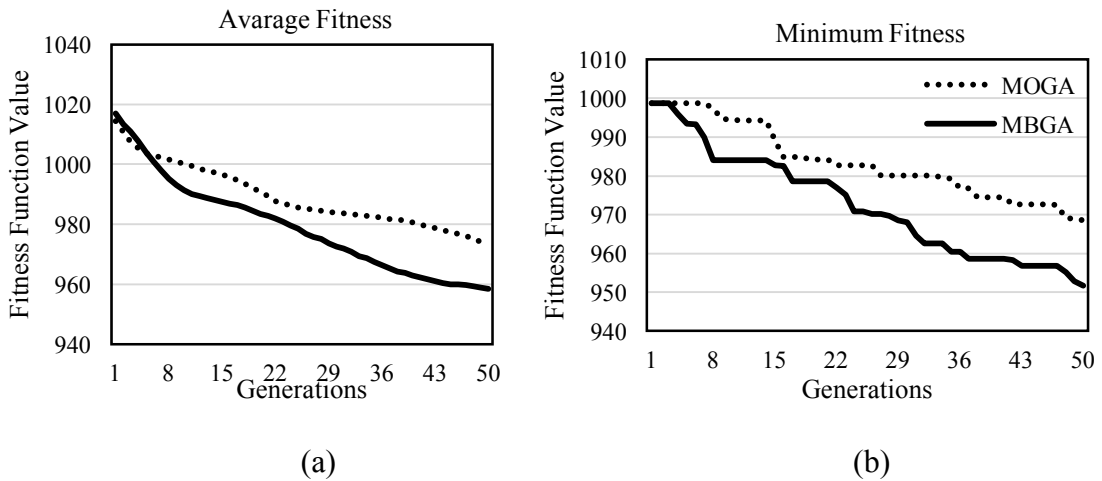


Figure 4. MBGA and MOGA for 100 jobs, $f(x)_{avg}$ comparison (a) $f(x)_{min}$ comparison (b).

When the plots are examined, it is seen that the proposed MBGA method had superiority over MOGA in terms of average fitness $f(x)_{avg}$ for each generation, as well as minimum fitness $f(x)_{min}$. The performance gap between MBGA and MOGA increased with the growth of the problem size. For all problem sizes, in terms of $f(x)_{min}$, MBGA reached a minimum at earlier stages. In terms of $f(x)_{avg}$, a significant increase in the average solution quality was observed, especially starting from the problem size of 50. As the size of the problem increased from this stage on, the beginning of the recovery of $f(x)_{avg}$ was directed towards the first generations.

TESTING THE PERFORMANCE OF THE PROPOSED ALGORITHM

A statistical analysis was performed to evaluate whether the developed Multi-MOORA Based Genetic Algorithm (MBGA) method was superior to the multiobjective genetic algorithm (MOGA). More than 30 trials were carried out by randomly updating the initial population consisting of 40 chromosomes for two methods on multiobjective flowshop scheduling problems consisting of 25, 50, and 100 jobs. In all trials, $f(x)_{min}$ values that were obtained along 50 generations were recorded. In order to select the method to be applied in the analysis of the results, the normality of the data was tested by using the normality test. Since the data were found to be in compliance with normal distribution, they were evaluated by applying paired samples t -test in the Minitab program. The test probabilities p and average of $f(x)_{min}$ values μ obtained by the test are shown in Table 2.

Table 2. Comparison of the results of the MBGA and MOGA obtained in multiple repetitive applications in the frame of p values.

	Problem Size					
	25		50		100	
	MBGA	MOGA	MBGA	MOGA	MBGA	MOGA
μ	228.7762	233.0434	458.9650	460.2724	949.5494	951.1581
p	0		0.043		0.038	

As seen in the results in Table 6, the result of $p \leq 0.05$ was obtained in all problem sizes in the study, and thus, MBGA was superior to MOGA. As a result of the statistical analysis, it was observed that the superiority of MBGA over MOGA was not accidental.

CONCLUSION

This study applied the Multi-MOORA method, which is a multicriterion decision-making method, to the solution of multiobjective flowshop scheduling problems by combining them with the genetic algorithm. The proposed Multi-MOORA-Based Genetic Algorithm (MBGA) method tries to optimize equally weighted C_{max} , \bar{F} , T_{max} , \bar{T} and NLJ criteria simultaneously. The Multi-MOORA-Based Genetic Algorithm was compared to the classical multiobjective genetic algorithm (MOGA) in terms of the minimum fitness $f(x)_{min}$ and the average fitness $f(x)_{min}$ for each generation of three differently sized problems, and it was found that the proposed algorithm excelled. With the proposed algorithm, the efficiency of genetic algorithms has been increased to solve NP-Hard class multiobjective flowshop scheduling problems. The results obtained support the Lin, Lee & Wu (2012) study, in which AHP was used with fewer criteria and applied in a different problem type. In addition, the study differs in terms of methodology and number of criteria, with the study in which Braglia & Grassi (2009) minimized makespan and maximum tardiness using NEH (Nawaz-Enscore-Ham) heuristic and TOPSIS. It accelerates the process of reaching a good solution in the integration of multicriteria decision-making techniques into multiobjective GA. However, since it contains extra code compared to the classical multiobjective GA, naturally, the time increases a little in terms of computation time. In future studies, the algorithm's performance may be improved by integrating the genetic algorithms of the Multi-MOORA method into different phases such as replacement, or the proposed MBGA method may be adapted to solve different problems.

Acknowledgment

This study was supported by the SAU Scientific Research Projects Commission (Project no: 2017-50-01-073).

REFERENCES

- Braglia, M. & Grassi, A. 2009.** A new heuristic for the flowshop scheduling problem to minimize makespan and maximum tardiness. *International Journal of Production Research* 47(1):273-288.
- Brauers, W. K. M. & Zavadskas, E. K. 2006.** The MOORA method and its application to privatization in a transition economy. *Control and Cybernetics* 35(2):445-469.
- Brauers, W. K. M. & Zavadskas, E. K. 2010.** Project management by MULTIMOORA as an instrument for transition economies. *Technological and Economic Development of Economy* 16(1):5-24.
- Brauers, W. K. M. & Zavadskas, E. K. 2012.** Robustness of MULTIMOORA: a method for multi-objective optimization. *Informatica* 23(1):1-25.
- Chachan, H. A. & Hasssanali, F. 2021.** Using Non-dominated Sorting Particle Swarm Optimization Algorithm II for Bi-objective Flow Shop Scheduling Problems. *Iraqi Journal of Science* 62(1):275-288.
- Chen, C. L., Vempati, V. S., & Aljaber, N. 1995.** An application of genetic algorithms for flow shop problems. *European Journal of Operational Research* 80(2):389-396.
- Das, S. R., Gupta, J. N., & Khumawala, B. M. 1995.** A savings index heuristic algorithm for flowshop scheduling with sequence dependent set-up times. *Journal of the Operational Research Society* 46(11):1365-1373.
- Deng, J. & Wang, L. 2017.** A competitive memetic algorithm for multi-objective distributed permutation flow shop scheduling problem. *Swarm and Evolutionary Computation* 32:121-131.
- Geng, K., Ye, C., & Liu, L. 2020.** Research on Multi-Objective Hybrid Flow Shop Scheduling Problem With Dual Resource Constraints Using Improved Memetic Algorithm. *IEEE Access* 8:104527-104542.
- Gonçalves, J. F., de Magalhães Mendes, J. J., & Resende, M. G. 2005.** A hybrid genetic algorithm for the job shop scheduling problem. *European journal of operational research* 167(1):77-95.
- Horn, J., Nafpliotis, N., & Goldberg, D. E. 1994.** A niched Pareto genetic algorithm for multiobjective optimization. *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence* 1:82-87.
- Ishibuchi, H. & Murata, T. 1996.** Multi-objective genetic local search algorithm. *Proceedings of IEEE International Conference on Evolutionary Computation* 119-124.
- Johnson, S. M. 1954.** Optimal two-and three-stage production schedules with setup times included. *Naval research logistics quarterly* 1(1):61-68.
- Keskin, K. 2010.** Beklemesiz akış tipi çizelgeleme problemlerinin çok amaçlı melez genetik algoritma ile çözümü. Doctoral dissertation, Selçuk Üniversitesi Fen Bilimleri Enstitüsü.
- Li, X. & Ma, S. 2016.** Multi-Objective Memetic Search Algorithm for Multi-Objective Permutation Flow Shop Scheduling Problem. *IEEE Access* 4: 2154-2165.
- Lin, D., Lee, C. K. M., & Wu, Z. 2012.** Integrating analytical hierarchy process to genetic algorithm for re-entrant flow shop scheduling problem. *International Journal of Production Research* 50(7):1813-1824.
- Marichelvam, M. K., Prabaharan, T., & Yang, X. S. 2014.** A discrete firefly algorithm for the multi-objective hybrid flowshop scheduling problems. *IEEE transactions on evolutionary computation* 18(2):301-305.
- Mokhtari, H., Abadi, I. N. K., & Cheraghlikhani, A. 2011.** A multi-objective flow shop scheduling with resource-dependent processing times: trade-off between makespan and cost of resources. *International Journal of Production Research* 49(19):5851-5875.
- Murata, T., Ishibuchi, H., & Tanaka, H. 1996.** Genetic algorithms for flowshop scheduling problems. *Computers & Industrial Engineering* 30(4):1061-1071.

- Pan, Q. K., Wang, L., & Qian, B. 2009.** A novel differential evolution algorithm for bi-criteria no-wait flow shop scheduling problems. *Computers & Operations Research* 36(8):2498-2511.
- Pasupathy, T., Rajendran, C., & Suresh, R. K. 2006.** A multi-objective genetic algorithm for scheduling in flow shops to minimize the makespan and total flow time of jobs. *The International Journal of Advanced Manufacturing Technology* 27(7):804-815.
- Ponnambalam, S. G., Jagannathan, H., Kataria, M., & Gadicherla, A. 2004.** A TSP-GA multi-objective algorithm for flow-shop scheduling. *The International Journal of Advanced Manufacturing Technology* 23(11-12):909-915.
- Poon, P. W. & Carter, J. N. 1995.** Genetic algorithm crossover operators for ordering applications. *Computers & Operations Research* 22(1):135-147.
- Pour, N., Tavakkoli-Moghaddam, R., & Asadi, H. 2013.** 5. Optimizing a multi-objectives flow shop scheduling problem by a novel genetic algorithm. *International Journal of Industrial Engineering Computations* 4(3):345-354.
- Rajendran, C. & Chaudhuri, D. 1990.** Heuristic algorithms for continuous flow-shop problem. *Naval Research Logistics (NRL)* 37(5):695-705.
- Robert, R. B. J. & Rajkumar, R. 2019.** A Hybrid Algorithm for Multi-Objective Optimization of Minimizing Makespan and Total Flow Time in Permutation Flow Shop Scheduling Problems. *Information Technology and Control* 48(1):47-57.
- Veeraiah, T., Reddy, P. Y., Kumar, P. V. S., & Milton, P. W. D. S. 2017.** Optimization of Flow Shop Scheduling by MATLAB. *SSRG International Journal of Mechanical Engineering (SSRG-IJME)* 222-226.