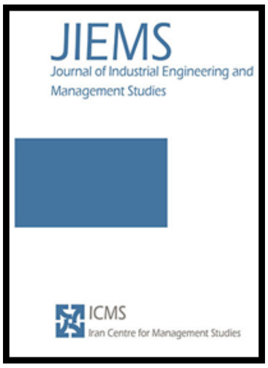
JIEMS

Journal of Industrial Engineering and Management Studies

Vol. 8, No. 1, 2021, pp. 45-71

DOI: 10.22116/jiems.2021.170958.1244

www.jiems.icms.ac.ir



Optimization of the multi-objective flexible job shop scheduling model by applying NSGAII and NRGA algorithms

Reza Ehtesham Rasi[1,*]

Abstract

Scheduling is one of the key parameters to maintain competitive advantage of organizations, and can directly affect productivity, reduce production time and increase the profitability of an organization. Job shop scheduling problem (JSSP) seeks to find the optimal sequence of performing various jobs related to group of machines. The purpose of this paper is to provide a multi objective to optimize makespan, energy consumption and machine erosion in flexible JSSP. The problem of this paper is to assign each operation to a machine and to order the operations on the machines, such that the maximal completion time (makespan) of all operations is minimized. The obtained model belongs to NP-Hard class of optimization problems. In terms of overcoming NP-hardness of the proposed model and solve the complicated problem, a non-dominated sorting genetic algorithm (NSGAII) is employed. As there is no benchmark available in the literature, the non-dominated ranking genetic algorithm (NRGA) is developed to validate the results obtained and test problems are provided to show the applicability of the proposed methodology and evaluate the performance of the algorithms. In this study, to evaluate the performance of these algorithms, they were statistically analyzed using T-test. Ultimately, results of the selected model were ranked by applying the technique for order of preference by similarity to ideal solution (TOPSIS).

Keywords: job shop scheduling problem; multi-objective; optimization; NSGAII, NRGA.

Received: February 2019-05

Revised: January 2021-05

Accepted: January 2021-09

1. Introduction

In competitive condition of markets, organizations need to optimize their efficiency and production process, so they have to consider optimization factors in their operation schedules to meet the demands of customers in a timely manner (Gamila and Motavalli, 2003). In manufacturing and industrial environments, it depends on factors such as the type of finished products, the amount of capital and available space, different production policies are adopted. Among these policies, job shop problem (JSP) is one of the most well-known and applicable

* Corresponding author; ehteshamrasi@qiau.ac.ir

[1] Department of Industrial Management, Qazvin Branch, Islamic Azad University, Qazvin, Iran.

one (Brucker and Schlie, 1990). Scheduling of resources leads to increased efficiency and utilization of capacity, reducing the time needed to complete the work and eventually increasing the profitability of an organization. Therefore, effective scheduling of machines and human resources in today's competitive environment is a necessity(Tavakkoli et al.,2010). In general, scheduling is referred to the activity of allocating resources to tasks over time. In the JSP model, *n* tasks must work on *m* machines by maintaining a certain sequence. The tasks are composed of various operations and processing on the machines is already known. The importance of flexibility is high where the researchers believe that the global competitive scene in today's production has seen organizations change from two indicators of productivity and quality to two indicators of flexibility and speed of operation. Therefore, the FJSP consists of two sub-problems of determining the sequences of tasks on machines and the allocations of the machines to operations (Kumar et al., 2012). By increasing the dimensions of the problem, traditional methods of determining the optimum answer lose their efficiency because they take a lot of time and the computation time increases exponentially. Along with the development of the industry, the issue of energy saving has been taken into consideration. The most developed countries around the world class, the issue of global warming, reduced fossil fuels and environmental degradation, and are seeking to cope with the recent effects of climate change and energy crisis. Consumption of energy like fossil fuels and electricity has many negative effects on the environment. That is why saving energy is one of the most important goals of scheduling in recent decades. Therefore, solving the problem of scheduling with the goal of reducing energy consumption can reduce costs, save energy and reduce greenhouse gas emissions. In addition, minimizing the cost of critical machine erosion can help the entire system. In fact, erosion instead of being assembled in a machine (a critical machine), spreads among all the machines which makes it possible for the critical machine not to erode and deteriorate. In this research, the scheduling of multi-objective flexible job shop has been studied taking into account three objectives. The researchers strived to present three-dimensional mathematical model to minimize the maximum completion time of tasks, total energy consumption of the system, and critical machine erosion in FJSP and then optimize it using two NSGAII and NRGA metaheuristic algorithms. Finally, MCDM technique was applied to select the appropriate and superior algorithm. This study contributes the literature by proposing multi-objective multi-period model to optimize the profit of the FJSP considering scheduling, in accordance to the other three objectives, minimizing the makespan, energy consumption and machine erosion. The obtained model belongs to NP-Hard class of the optimization problems, thus NSGA-II and NRGA are developed to find a near-optimum solution. This study is structured as follows: in the second part, the literature on the subject is examined. In the third section, the mathematical model of optimization of the maximum time of completion of tasks, the total energy consumption of the system and the critical machine erosion in FJSP is introduced; also, parameters, objective functions, and related constraints are described. In section 4, the multi-objective optimization algorithms NSGAII and NRGA are implemented, in section 5, the two algorithms are compared using five indicators and one MCDM technique, and finally an effective algorithm is introduced. In section 6, we presented managerial insight and the conclusions and suggestions for future research are presented in section 7.

2. Literature Review

Job shop scheduling or job-shop problem (JSP) is an optimization problem in computer science and operations research in which jobs are assigned to resources at particular times. The most basic version is as follows: We are given n jobs J_1, J_2... J_n of varying processing times, which need to be scheduled on m machines with varying processing power, while trying to minimize the makespan. The makespan is the total length of the schedule. The FJSP is an extension of the classical job shop-scheduling problem, which allows an operation to be processed by any machine from a given set. Brandimarte (1993) was the first person to use decomposition in the optimization of FJSP. Gambardella and Mastrolilli (1996) designed Tabu Search algorithm for FJSP and improved Tabu Search algorithm. In their study, they presented two new neighborhoods, and studied 178 FJSP as well as 43 job shop problems. In this regard, Deb et al. (2000) presented NSGA-II, one of the most efficient and popular multi-objective optimization algorithms. Deb et al. (2002) proposed multi objective evolutionary algorithms (EAs) that use non-dominated sorting for the first time. Kacem et al. (2002a) proposed an integrated approach and GA to solve this problem. They also used a chromosomal view that combined routing, sequencing data and designed an approach called "localization" to find appropriate initial allocations, and the distribution rules are used to determine the sequence of operations. Based on local optimization principles, Kacem et al. (2002b) presented multi-objective FJSP for three common goals: minimizing completion time, minimizing workload maximization on machines in terms of time, and minimizing the total workload of machines by combination of fuzzy logic and evolutionary algorithms. Xia and Wu (2005) presented a hierarchical application based solution to solve the problem of MOFJSP. The proposed method used Particle Swarm Optimization (PSO) method for assigning operations to machines and Simulated Annealing (SA) algorithm for the proper sequence of operations on the machines. Liu et al. (2006) proposed a hybrid algorithm from the integration of two variable neighborhood search algorithm and PSO, for MOFJSP mode. Gao et al. (2007) proposed the scheduling of FJSP based on three goals: minimizing completion time, minimizing the workload on machines in terms of time, and minimizing the total workload of machines. They designed a hybrid GA for the problem. In order to enhance the search ability, people in the population are promoted through declining neighborhood approach that includes two local search processes: local search by moving an operation, and local search by moving two operations. Al Jadaan et al. (2008) proposed method combining the new Ranked based Roulette Wheel selection algorithm with Pareto-based population ranking Algorithm, named NRGA which alleviates most of the above three difficulties. A two tier ranked based roulette wheel selection operator was presented that creates a mating pool from the parents' population by selecting the best solutions stochastically. Simulation results on benchmark test problems showed that the proposed NRGA, in most of the problems was able to find much better spread of solutions and faster convergence near the true Pareto-optimal front compared to NSGA-II other elitist MOEA that pay special attention to creating a diverse Pareto-optimal front. Much better performance of NRGA is observed. Frutos et al. (2010) used Memetic algorithm based on NSGAII algorithm, which also adds a local search method (Simulated Annealing) to solve its goals, which is the total operating cost and time. Lei (2011) developed a Simplified Multi-Objective Genetic Algorithm for FJSP with the goal of minimizing completion time and total latency ratio at the same time. Rahmati and Zandieh (2012) proposed a new population-based evolutionary algorithm, called Biogeography-Based Optimization Algorithm (BBO) for this problem solving. The main purpose of their paper was to present a new way for BBO to resolve scheduling problems. They also compared this algorithm with the newly developed GA algorithm that have similar operators, taking into account three goals of completion time, critical load, and total workload of machines. Li and Pan (2012) presented a Discrete Chemical-Reaction Optimization (DCRO) algorithm to solve FJSP with and without preventive maintenance constraints. Three proposed objectives included completion time, the workload of the critical machine and a chemical molecule represents total workload for each solution. Xiong (2013) proposed a robust scheduling for FJSP with machine failure randomly based on FJSP under uncertainty conditions named stochastic FJSP (S-FJSP). Jiang et al. (2014) proposed the optimization of multi-objective flexible job shop for completion time, processing cost, energy consumption and cost-effective processing. Then they proposed a Blood-Variation-based NSGA-II (BVNSGA-II). They proposed this algorithm by optimizing crossover and NSGA-II mutation in order to overcome the population prematurity. Ahmadi et al. (2016) evaluated an optimization methodology for two objectives of completion time and stability in a random environment to deal with machine failure in the FJSP. In this study, the stable scheduling in FJSP with machine failure is shown randomly. Thus, a multi-objective approach to the FJSP is presented in machine failure conditions, with two evolutionary algorithms NSGA-II and NRGA. Shen et al. (2018) addressed the FHSP with sequence - dependent setup times and where the objective is to minimize the makespan. Movahed & Nama (2019) derived a closed-form solution for optimal production and capacity under dedicated and flexible policy when demand and lead-time follow uniform and normal distribution. Goli et al. (2019) provided that scheduling is known as a great part of production planning in manufacturing systems. This paper either addressed a novel robust FSS problem with outsourcing option where jobs can be scheduled for inside or outsourced to one of the available subcontractors. The objective is to minimize the total-weighted time required to complete all jobs and the total cost of outsourcing. Therefore, a Robust Mixed-Integer Linear Programming (RMILP) model was proposed to accommodate the problem with the real-world conditions. Tirkolaee et al. (2019) investigated a novel fuzzy multi-objective multi-period Aggregate Production Planning (APP) problem under seasonal demand. The main goals were to minimize the total cost including in-house production, outsourcing, workforce, holding, shortage and employment/unemployment costs, and maximize the customers' satisfaction level. Hosseini (2019) has been studied the two-stage assembly flow shop scheduling problem. He supposed that number of products of different kinds is needed to be produced. There are m uniform machines in the first stage to manufacture the components (parts) of products and there is one assembly station in the second stage to assembled parts into products. Set up operation should be done when a machine starts processing a new part and setup times are treated as separate from processing times. Two objective functions were considered: 1- minimizing the completion time of all products (makespan) as a classic objective, and 2- minimizing the cost of energy consumption as a new objective. Li et al. (2020) studied the analysis of multi-objective FJSP, a multi-objective low-carbon job-shop scheduling problem (MLFJSP) with variable processing speed constraint is proposed in this paper. The optimization objectives of MLFJSP include minimizing the makespan, total carbon emission and machine loading. Meanwhile, an improved artificial bee colony algorithm (IABC) is designed to solve the MLFJSP. Two main components of multi-objective algorithms are allocation of fitness to the members of the population based on the sorting of unsatisfactory responses; and the preservation of density among the similar non-dominated recessive queues. However, researchers believe that this area needs to introduce useful operators to solve multi-objective optimization problems more efficiently. Particularly, attention to the introduction of elitism to enhance the convergence characteristics of multi-objective evolutionary algorithm is one of the most important issues. Elitism has an impact on the achievement of better convergence of multi-objective algorithms. In this paper, we strive to optimize the MOFJSSP through three-objectives of the energy consumption, critical machine erosion, and maximum completion time of tasks simultaneously. The problem of this paper is to assign each operation to a machine and to order the operations on the machines, such that the maximal makespan of all operations is minimized.

Table 1. Background of research

Authors	Objective function	Model Purpose	Definition of problem	Solution method	Criteria
Ahmadi et al (2016)	Bi objective	The time for completing the tasks Stability	Stable scheduling for FJSP by considering machine breakdowns randomly	Genetic Algorithm for NGSA II, NRGA	Makespan, machine failure
Shen et al (2018)	Bi objective	Sequence dependent setup times were considered	Solving the flexible job shop scheduling problem with sequence-dependent setup times	Tabu search algorithm	setup times, Makespan
Hosseini (2019)	Bi objective	minimizing the completion time of all products and the cost of energy consumption	A bi-objective model for the assembly flow shop scheduling problem with sequence dependent setup times and considering energy consumption	Epsilon constraint algorithm	Makespan, energy consumption
Li et al (2020)	Bi objective	The optimization the makespan, total carbon emission and machine loading	A multi-objective low-carbon job-shop scheduling problem (MLFJSP) with variable processing speed constraint	an improved artificial bee colony algorithm (IABC)	Makespan, carbon emission machine loading
Jiang & Wang (2020)	Multi objective	minimization both makespan and total electricity cost (TEC) simultaneously	decrease the load of electricity grid during the peak period; time-of-use electricity price	hybrid multi-objective evolutionary algorithm	Makespan, total electricity cost
This paper	Multi objective	Optimization energy consumption, critical machine erosion, and maximum completion time of tasks simultaneously	The problem of this paper is to assign each operation to a machine and to order the operations on the machines, such that the maximal completion time (makespan) of all operations is minimized	NSGAII & NRGA	Energy consumption, erosion, Makespan

3. Problem definition and model formulation

In this section, the problem is defined and indices, parameters, and decision variables of the model as well as model formulation have been presented.

3.1. Problem definition

FJSP involves number of specific operations that are assigned to machines in accordance with the input parameters of the problem. Therefore, the goal is to allocate tasks to machines, taking into account the simultaneous scheduling, so that the total makespan, the critical machine erosion, and the energy consumption of the system are also minimized. In this research, the objective function of FJSSP is considered as multi-objective, and we will add two objectives of minimizing energy consumption and the erosion of critical machine in conventional and time-consuming task. The flexibility factor in job shop problem is considered because flexibility is the ability of a system to adapt to various variations and the ability of a system to deal with changes. Today, rapid production environments are known for short life span and increased product diversity. In such environments, product flexibility is a key of competitive weapon, and it makes the factory be able to respond to unpredictable changes (such as a change in customer needs) as the main factor for competitive pressure. The assumptions of MOFJSSP mathematical model is as follows:
(1) Each job consists of a sequence of operations that allows them to be processed on any available machine;
(2) All jobs and machines are available at zero time;
(3) At any given time, only one operation can be performed on any machine;

(4) Discrete operations are not allowed, each operation must be completed without interruption;
(5) Machine preparation and transposition times between operations can be neglected.
Before describing the mathematical model of the indices, it is better to define the parameters and decision variables as follows:

3.2. Indices, parameters, and variables of the model

The following indices, parameters, and variables are used in modeling MOFJSSP:

Table 2. The parameters of the proposed FJSSP model

i,h : Index of The counter of jobs, $1,2,\ldots,n$	β_k : Coefficient of changing processing time to consumption of energy machine k
j,g : Index of Operation counter, $1,2,\ldots,J_i$	E_{ijk} : Energy consumption of the machine k to complete operation j of job i
k : Index of The counter of machines, $1,2,\ldots,m$	$C_{\max}$ :: Maximum time for completing all jobs
n : Total number of jobs	C_{ij} : Completion time of O_{ij}
m : Total number of machines	C_{hg} : Completion time of O_{hg}
J_i : Total number of operations to complete job i	X_{ijk} : If O_{ij} is done on the machine k, it must be 1, otherwise zero
O_{ij}: Operation j of job i	X_{hgk} : If O_{hg} is done on the machine k, it must be 1, otherwise zero
P_{ijk} : processing time of O_{ij} if to be done on machine k	Z_{ijhgk} : If the O_{ij} operation is preceded by the O_{hg} on machine k, it must be 1, otherwise zero.
P_{hgk} : processing time of O_{hg} if to be done on machine k	Z_{hgijk} : If the O_{hg} operation is preceded by the O_{ij} on machine k, it must be 1, otherwise zero
α_k : Erosion factor machine k	

3.3. Model formulation

The model has three objective functions and ten constraints.

$$Z_1 = min\{Cmax\} \tag{1}$$

$$Z_2 = Min\sum_{i=1}^{n}\sum_{j=1}^{J}\sum_{k=1}^{m} E_{ijk}X_{ijk} \tag{2}$$

$$Z_3 = Min\left\{Max\left[\sum_{i=1}^{n}\sum_{j=1}^{J} a_k p_{ijk}X_{ijk}\right]\right\} ;\forall i,k,\forall j = 2,\ldots J \tag{3}$$

$$E_{ijk} = B_k P_{ijk} \tag{4}$$

$$C_{ij} - C_{ij-1} \geq P_{ijk}X_{ijk};\forall i,k,\forall j = 2,\ldots,j_i \tag{5}$$

$$C_{ij} \geq P_{ijk}X_{ijk};\forall i,j = 1,k \in m \tag{6}$$

$$\left(C_{hg} - C_{ij} - P_{hgk}\right)X_{hgk}X_{ijk}Z_{ijhgk} \geq 0;\forall i,h,j,g,k \in m_{ij} \cap m_{hg} \tag{7}$$

$$\left(C_{ij} - C_{hg} - C_{ijk}\right)X_{ijk}X_{hgk}Z_{hgijk} \geq 0;\forall i,h,j,g,k \in m_{ij} \cap m_{hg} \tag{8}$$

$$\sum_{k \in mij} X_{ijk} = 1;\forall i,j \tag{9}$$

$$Z_{ijhgk} + Z_{hgijk} = X_{ijk}X_{hgk};\forall i,j,h,g,k \in m_{ij} \cap m_{hg} \tag{10}$$

$$C_{max} \geq C_{ij} \tag{11}$$

$$C_{ij} \geq 0; \forall i, j \tag{12}$$
$$X_{ijk} \in \{0,1\}; \forall i, j, k \tag{13}$$

The first objective represents the minimization of the maximum completion time of tasks i.e. the jobs allocated to the machinery was carried out in the shortest possible time. The second objective is the minimization of TEC or the total energy consumed by the system, this objective shows the relationship between processing time and energy consumption. The third objective is the minimization of critical machine erosion (CME). This objective function disperses erosion rather than aggregation into a machine (critical machine) on all machines. Finally, these three objectives are looking to optimize the scheduling of flexible job shop production system. Constraint (4) shows the relationship between the processing time and the energy consumed. Constraint (5) as the first constraint expresses the position that each work follows the sequence of the specified operation. Constraint (6) shows that the completion time of the first operation *i* is equal to the minimum processing time of O_{ij}. Constraints (7 - 8) indicate that O_{hg} should not start before the O_{ij} operation ends, or O_{hg} must be completed before the O_{ij} operation begins, if both are assigned to the same machine K. Constraint (9) shows that for each operation, a machine must be selected from available machines. In other words, each O_{ij} must be done only on one machine. Constraint (10) states that one of the two-priority constraints must be selected. Constraint (11) shows that the maximum time for completing all jobs is larger than or equal to the completion time of O_{ij}.Constraints (12-13) also indicate the gender of the variables.

4. The solution method

Most real world optimization issues have a place with the class of NP-hard issues. Habitually, precise techniques cannot tackle this class of issues in typical and sensible time. This class of meta-heuristic is reasonable devices. Frequently, exact methods cannot solve this class of problems in normal and reasonable time. To optimize this class of meta- heuristic algorithms are suitable tools. For experimentation, this section presents the application of the proposed model along with the proposed NSGA-II algorithm on some random test problems. Some sets of small, medium and large sized instances are considered to evaluate the performance of the solution approach. In addition, since no benchmark is available in the literature to verify and validate the results obtained by NSGA-II, another popular MOEA called NRGA is suggested to solve the problem as well. To do this, some preliminary concepts and principles of NSGA-II and NRGA are first reviewed. Then, the required structures, i.e. chromosome encoding and decoding, crossover and mutation operators are described. A multi-objective genetic algorithm based on the NSGA-II is utilized to provide Pareto fronts of the conflicting objectives. The purpose of employing the second algorithm (NRGA) is to verify the results obtained by NSGA-II. It is noticed that since most real world cases are large scale and NP-hard, it justifies the use of meta-heuristics; however, they do not guarantee to achieve optimal solutions and usually obtain near optimal solutions.

The FJSP in this research goes under the rubric of NP-hard category, which cannot be solved by applying exact methods. To solve such problems, heuristic and meta-heuristic methods based on optimization of hybrid problems were applied. In this study, for validation, valid mathematical models presented by leading researchers in mathematical modeling of job shop problems have been used as the basis of the model design. In order to solve the MO-FJSSP mathematical model, the meta-heuristic NSGA-II and NRGA algorithms are utilized, which have been one of the most widely used multi-objective optimization methods in recent years. At the end, the efficiency of these two algorithms is compared and the superior algorithm introduced according to Ahmadi et al. (2016) research.

4.1. NSGA-II algorithm

Multi-objective evolutionary algorithms (MOEAs), which are classified as one of Meta heuristic algorithms, are also used to solve multi objective optimization problems. In this paper, among different MOEAs algorithms, the NSGA-II and NRGA are used for solving MO-FJSSP, with an efficient procedure to generate the initial population. In NSGA-II algorithm, the chromosomes originally existing in the initial population P_t are used to produce new chromosomes by utilizing coupling agent. NSGA-II is one of MOEA techniques that optimize the problem objectives simultaneously, without being affected by any other solution. The computational complexity of NSGA algorithm stimulate from existing complexity of a non-dominated sorting procedure in every generation, results in an expensive procedure as compared to NSGA-II algorithm for large population sizes. New chromosomes form Q_t population, and then two populations merge and generate R_t population with 2N chromosomes that is called elitism. Non-post ranking is applied to the R_t population. The responses located on the fronts up to the level required to supply N chromosomes in the population are selected and other chromosomes are eliminated. In many cases, it may be possible to create conditions where the last one has more chromosomes of the number needed to complete the population; in such condition, number of chromosomes should be eliminated. For this purpose, it is advisable to use crowding distance and select chromosomes that are more diverse. Selection in this algorithm is performed by binary tournament selection. In this way, two members of the population are randomly selected, two members are compared and any one, which is better in terms of the criteria mentioned in the next stage, is selected.

4.1.1. Selection criteria in the NSGA-II algorithm

Step 1: Non-dominated Sorting

To determine the quality of each member of the population, non-dominated sorting is used, so that the value of the target function is computed for each member, each member that is not dominated by any member is ranked first and forms the first front.

$$\begin{cases} \forall i : x_i \le y_i \\ \exists i_0 : x_{i_0} \prec y_{i_0} \end{cases} \quad \longleftrightarrow \quad \text{X dom Y}$$

Y is not better than X for any purpose.
X is absolutely better than Y in at least one target.

By identifying the members of the first front, these members will be set aside and members of the second front will be appointed, which will continue until all members are ranked.

Step Two: Crowding distance

The selection of the members of the main population begins with low ratings. When the number of remaining members has not yet been determined, the lower-ranking members are less likely to be selected. Among those members with similar ranks, the ones are selected which have more crowding distance. The crowding distance between each member is the sum of its overlaps in relation to each objective. The propagation range for each objective is the ratio of the distance between the previous and next member of the target member to the distance between the minimum and the maximum value created by the members for that target. The greater distance is desirable because of the variety of answers. Fig (1) and equation (14) show calculation of population distance.

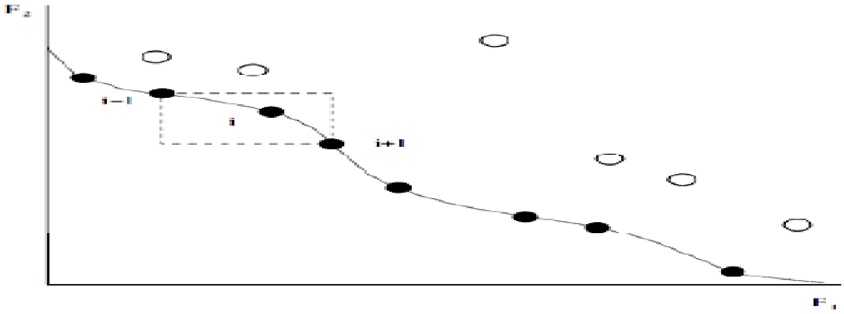


Figure 1. Calculation of parameter called population distance (Deb et al., 2000)

$$d_i = \frac{|f_1^{i+1} - f_1^{i-1}|}{f_1^{\max} - f_1^{\min}} \tag{14}$$

4.1.2. The steps of the NSGA-II algorithm

At first, an initial population P_0 is randomly generated. The frontier is ranked, and the chromosomes of each frontier are matched to its equivalent rank. For example, the best chromosomes placed on the surface will have fit one. By applying the crossover and mutation operators, the Q_0 population is generated. The other steps are as follows:

1. Combine parents and children populations together. $R_t = P_t \cup Q_t$
2. Apply the non-post rating to R_t population and classify the chromosomes on the F_i fronts.
3. Create a new population $P_{t+1} = \varnothing$. Assume the counter *i* equal to one.
4. As long as $|P_{t+1}| + |F_i| \prec N$, perform the operation $P_{t+1} = P_{t+1} \cup F_i$ and do *i = i +1*.
5. Run the rating based on the crowding distance $(F_{i,} \prec C)$ and place the most scattered answers $(N - |P_{t+1}|)$ from the solutions sorted in F_i on P_{t+1}.
6. Create new chromosomes in the population Q_{t+1} by applying binary operators based on the crowding distance, crossover, and mutation on the population P_{t+1}.

In the fifth step, using the crowding distance parameter, the ranking of chromosomes located on the front (non-post surface) *i*, which cannot fully appear in the population. Based on this parameter, the chromosomes are ranked in descending order, and then the binary selection operator is applied based on the crowding distance. The strong point of NSGA-II is to use crowding distance to maintain the diversity of responses located on the Pareto front; however, calculating the crowding distance and classifying chromosomes in different fronts adds significant computational load to that algorithm.

4.1.3. Termination condition

The termination condition is 100 repetitions, fig (3) depicts whole of the process.

4.1.4. Chromosome representation

Chromosome representation is an important issue for the GA in regards of computational time. It can be concluded from the works of Ho et al. (2007) that the search space of an operation-based representation covers the whole solution space and any permutation of operators can correspond to a feasible schedule. In this paper, the permutation-based chromosome representation proposed by Kacem et al.(2002a) has been used. This permutation-based chromosome composes of a matrix whose each row consist of triples:
(j, i, k) forms the chromosome, in which the

- j is current job number.
- i is operation number within job j .
- k is machine assigned to the operation.

4.1.5. Initialization

For the underlying populace, we utilized Ini-PopGen heuristic recommended by Al-Hinai and ElMekkawy(2011) that arbitrarily allocates the priority to jobs. Consequently, an activity is allocated to the machine, which can be done with preparing rather than the remainder of suitable machines. This system considers preparing time and outstanding burden on each machine.

4.1.6. Genetic operators

Accomplishing elite performance of GA is profoundly dependent on the performance of GA that is utilized in these algorithms (Gen and Cheng, 2000). In this manner, utilizing fitting administrators is an essential factor for broadening GA. Therefore, performing appropriate operators is infeasible algorithm. In the present circumstance, we need to play out a maintenance component that could be tedious. In this manner, it is more practical to design the operators that keep up the practicality of the timetable and avoid the maintenance components. Selected operators consist of precedence preserving order-based crossover (POX) and modified Position Based Mutation (PBM) as well as Machine Based Mutation (MBM) for performing the mutation. The main advantage of these operators is that no infeasible chromosome is produced.
All algorithms are coded and executed by using MATLAB R20179a on an Intel® Core™ i5 CPU @ GHz with 8 GB RAM, 64 Bit operating system. The chosen parameter values for both NSGA-II and NRGA algorithms are as follows: N pop is 100, crossover and mutation’s rates are 0.8 and 0.3 respectively, and the number of gene for mutation is considered 3.Nsim is determined based on experience in which the value of 10 results in higher value. Stopping criteria is determined based on the number of generations, which is chosen as 200. Taguchi experimental design method has been used to adjust parameters of the NSGAII algorithm. Taguchi method of criterion (S / N) use this criterion shows the amount of changes that have occurred in the response variable. For each factor, the optimal surface value is that the standard value (S / N) is higher; therefore, according to Figure (2) for all four factors, the results of parameter adjustment are as follows:

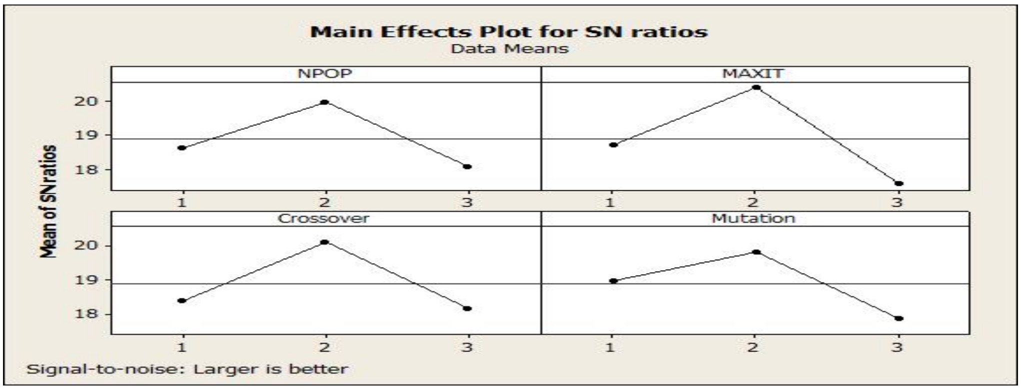


Figure 2. S/N rate for coefficients of ant algorithm

4.1.7. Non-dominated ranking genetic algorithm

Among of all type of MOEAs, MOEA algorithms in class of Pareto-based approach are the most appropriate algorithms in which is emphasize is on moving toward the true Pareto-optimal region in the selection process. According to several studies, the two well-known algorithms of this class are NRGA and NSGA-II, which are used in this paper. The simulation results on benchmark test problems show that both of these algorithms have outperformed many of the classical state-of-the-art algorithms. Therefore, using both of the MOEAs is critically appropriate in the coming study of the MO-FJSP.

4.2. NRGA algorithm

Another meta-heuristic algorithm that can be used to solve multi-objective problems to determine Pareto responses is NRGA algorithm. The function of this algorithm is similar to NSGA-II algorithm at various stages. The main difference is in the selection stage, in which the roulette wheel is used instead of using the binary match operator. As previously explained, the selection is based on the probability in a roulette wheel, and chromosome in fitness. In this algorithm, like in NSGA-II algorithm, fitting is based on two criteria of rank and crowding distance. First, a rank is selected from among the rankings using a roulette wheel. At this stage, lower ratings are more likely to be chosen.

$$P\{F_1\} \geq P\{F_2\} \geq P\{F_3\} \geq \dots$$

By reusing the roulette wheel from among the members present at this rating, the parent is determined that in this cycle, the selection of members with more crowding distance is more likely to occur.

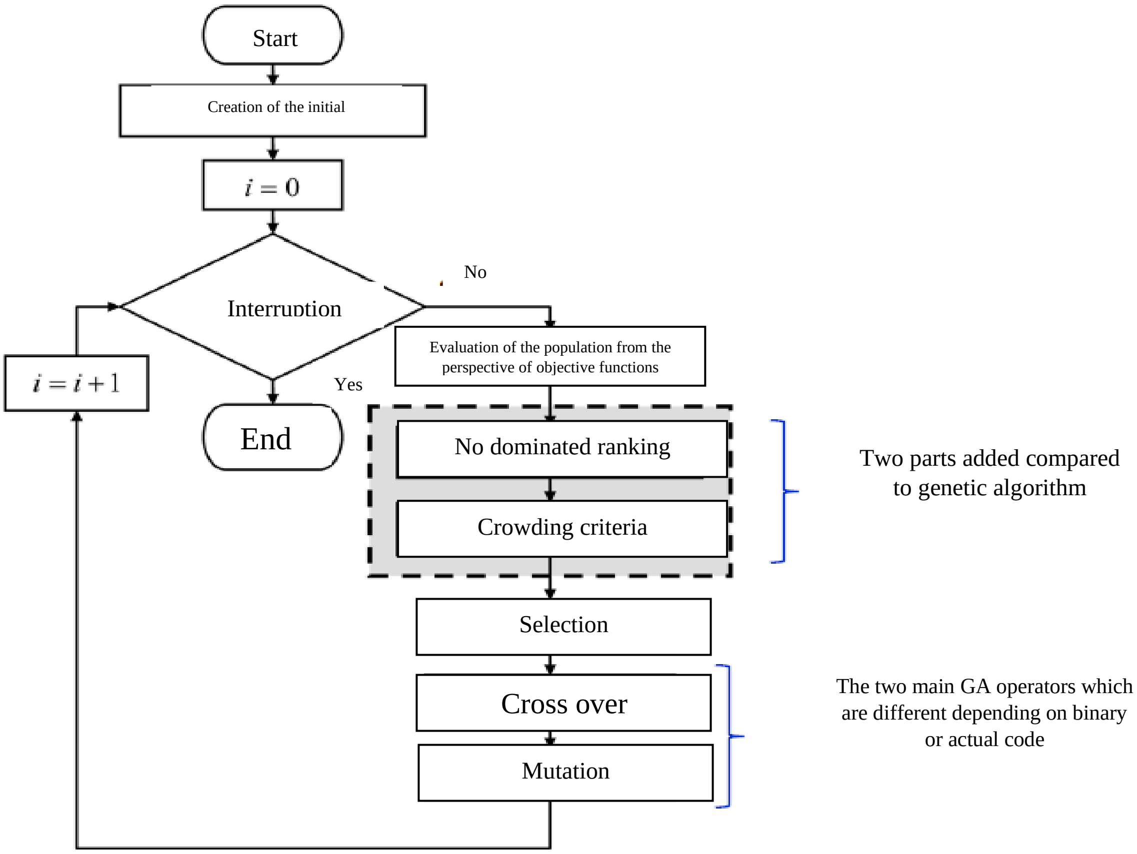


Figure 3. Flowchart of NSGA-II algorithm (Deb et al., 2000)

5. Criteria for comparison of multi-objective algorithms

In this paper to solve the multi-objective mathematical model used an algorithm deals with two conflicting objectives, including convergence Pareto optimal solutions and variation among the set of solutions that have two distinct and somewhat opposite objectives in multi-objective evolutionary algorithms (MOEA). Therefore, the best solution method used to solve the model, which is close to the real optimal solution with the same uniformity throughout the Pareto optimal region. If we can find a criterion by which we obtain the closeness of the final solutions to Pareto optimal solutions, and another criterion by which we can determine the density of the solutions, then we can measure the performance of two algorithms. In this research, five evaluation criteria have been investigated.

5.1. The Number of Pareto solution (NPS)

The NPS value indicates the number of Pareto optimal solutions that can be found in each algorithm. The more noteworthy the quantity of Pareto solution is favorable.

5.2. The mean ideal distance (MID)

This criterion is used to calculate the mean distance of Pareto solutions from source coordinates. The lower this criterion will be the more efficient the algorithm.

$$MID = \frac{1}{NPS}\sum_{i=1}^{NPS} c_i$$

$$where c_i = \sqrt{\sum_{j=1}^{m} f_{ji}^2}$$

5.3. The maximum spread or diversity

This criterion presented by Zitzler (1999), measures the length of the cube diameter of the space used by the final values of the objectives for non-dominated solutions set. For example, in the two-objective mode, this criterion is equal to the Euclidean distance between the two boundary solutions in the target space. The larger the size will be the better the solution.

$$D = \sqrt{\sum_{j=1}^{m} (\max_i f_i^m - \min_i f_i^m)^2}$$

5.4. Spacing criterion (S)

This criterion is the relative distance between successive responses. The measured value is equal to the smallest sum of the magnitude of the difference in the values of the objective functions between i^{th} solution and solutions in the final non-dominated set. This criterion measures the standard deviations of different values. When the answers are uniformly aligned, then the value of S will also be small, so the algorithm with its finely tuned small-numbered solutions will be better.

$$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} (d_i - \bar{d})^2}$$

$$d_i = \min_{k \in n, k \neq i} \sum_{m=1}^{2} | f_m^i - f_m^k |$$

$$\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$$

5.5. Computational time (CPU time)

The implementation time of the algorithms is one of the most important indicators in the efficiency of any meta-heuristic algorithm that can be used to compare the two algorithms NSGA-II and NRGA.

5.6. Implementation the algorithm

In this research, first, ten problems are produced for testing the performance of two multi-objective algorithms. The data needed to generate a problem include the number of machines (M), the number of jobs (N), and the maximum number of operations per job. The first five samples generated, whose solution time is less than 200 seconds are considered small dimensional samples, and the second 5 samples, whose solution time is more than 200 seconds by the software are called large dimensional samples. Each algorithm is coded using MATLAB software (R2016a), running on Windows7 and on the Intel® Core ™ 2 Due CPU laptop. The data generated by the algorithms is presented in table (3). Samples 1 to 5 are the samples that have been solved by software within less than 200 seconds and considered as samples of small dimensions. Samples 5 to 10 are samples of large dimensions that have been solved by software over 200 seconds.

Table 3. Different samples

Sample	Machine(M)	Job(N)	Operation(NO)
1	3	6	11
2	5	8	22
3	8	12	32
4	8	20	75
5	10	25	71
6	20	50	728
7	25	55	576
8	30	60	669
9	35	65	797
10	40	70	940

In the first sample, there are 3 machines, 6 tasks and 11 operations in which the processing times are presented in table (4).

Table 4. Processing times for the first sample

		Processing times		
		M_1	M_2	M_3
J_1	O_{11}	12	11	19
J_2	O_{21}	6	16	16
J_3	O_{31}	8	2	11
	O_{32}	12	2	16
J_4	O_{41}	inf[2]	12	1
	O_{42}	3	10	7
J_5	O_{51}	4	7	7
	O_{52}	16	11	13
J_6	O_{61}	6	15	inf
	O_{62}	14	inf	19
	O_{63}	inf	2	4

Given the data from samples 1 to 10, the model was solved with NSGA-II and NRGA algorithms. In both algorithms, the termination condition is 100 repetitions and population of 100 is considered. The crossover coefficient and the mutation are also considered to be 0.8 and 0.3, respectively. Table (4) shows the results of software run on the criteria used to evaluate two algorithms by applying NSGA-II; table (5) shows results after applying NRGA. In bellow figure, alleles on chromosome in the "main population" represent the stations while the alleles in its "affiliated population" represent the line labels, and the stations that belong to the same line are connected sequentially. An individual, which is a characteristic entity of

[2] Infinity

a chromosome, represents a network scheme. If a station appears multiple times in a chromosome of “main population”, it represents a transfer station. The chromosomes in the same population have different lengths because of different number of transfer stations in networks. In the iteration process, the chromosomes in the “main population” and its “affiliated population” whose lengths are dynamically changed have one-to-one correspondence. For large network, it is difficult to generate a large number of initial solutions with high diversity under strict constraints. At present, the main initial solution generation procedures can be classified as follows. Random connection method of adjacent nodes. In this method, the routes are generated by the connection between adjacent nodes, such as the probability-based IRSG procedure proposed by Jha et al. (2019). The method usually has high calculation efficiency, but cannot guarantee that the travel time between OD pairs will not be too long along the routes.

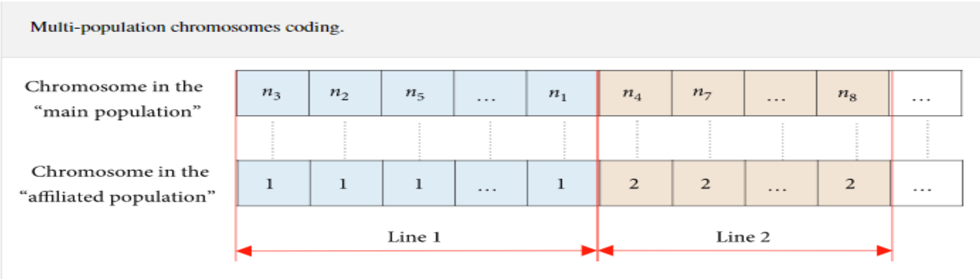


Figure 4. Multi population chromosomes coding

Each chromosome consists of three segments, in which the segments are used to demonstrate the relationship among flexible job shops. The first part with a dimension of [(S + F) × R] represents the relationship between in each period which are given as integer numbers belong to 1 to [(S + F) × R] as a priority value. Production of products in different periods and distribution quantities of the products from. In the indirect encoding approach, the metaheuristic handles a permutation of items, this is passed to greedy decoder that builds a solution for the problem. One advantage of this approach is that all problem-specific information is contained within the decoder, whilst the metaheuristic is kept canonical. In order to assess the performance of our different encoding/decoding strategies, two population-based metaheuristics are used. The first is the genetic algorithm NSGA-II. There are two main reasons for the choice of NSGA-II as a benchmark algorithm. First, the algorithm is well known due to its inclusion in many free and commercial software packages and to its performance in solving a large variety of multi-objective optimization problems. Second, NSGA-II often outperforms two other competitive grounded multi-objective evolutionary algorithms, namely IBEA and SPEA2. The results are reported on a set of random selected test instances. The second benchmark algorithm is the multi-objective Pareto local search PLS, which proved its ability of obtaining a good approximation of the efficient set in several multi-objective problems. We embedded the encoding/decoder strategies described above in both metaheuristics and compared their performance in solving.

After assessing the performance of the proposed encoding/decoding strategies embedded in multi-objective metaheuristics, we analyze the effectiveness NRGA and NSGA-II using the best encoding/decoding strategy (ED) in solving MO. The generated approximated Pareto fronts from the different metaheuristics on four randomly chosen instances are plotted too. Although the selected instances have the same sizes, they are not equally difficult to solve. This is what led to different frontiers shapes and affected the number of the generated potentially efficient solutions of each algorithm.

Table 5. The outputs of comparison criteria for NSGA-II Algorithm

sample	NSGA-II				
	↑ D	↓ MID	↓ Time	↓ S	↑ NPS
1	18.4932	199.892	146.057457	1.8856	3
2	28.1069	255.4597	154.480240	1.1606	7
3	31.7648	229.8202	155.458579	4.4091	5
4	88.1419	1169.5003	151.023812	7.5087	11
5	25.4915	820.1451	153.142928	0.4714	6
6	315.8781	19153.6413	226.234025	22.2418	13
7	581.1686	14500.9531	230.914141	99.5187	13
8	667.1686	17547.6028	218.901540	13.9134	30
9	235.1531	20696.4567	238.039755	10.441	18
10	453.0905	25326.7251	277.335522	28.4035	25
Mean	244.4582	9990.01963	195.1587999	18.99538	13.1

Table 6. The outputs of comparison criteria for NRGA Algorithm

sample	NRGA				
	↑ D	↓ MID	↓ Time	↓ S	↑ NPS
1	18.4932	199.892	150.135231	1.8972	3
2	39.9375	255.3479	151.668254	1.1702	6
3	43.3705	237.4294	159.422171	5.4091	5
4	30.8221	973.929	150.909716	7.5087	4
5	30.2159	743.2552	155.459260	1.4714	5
6	606.5328	18995.0544	225.703091	22.2428	30
7	631.8607	14824.132	282.407949	99.5654	27
8	670.7995	17348.7243	211.847407	14.9734	25
9	816.2236	21150.0958	221.063138	11.255	31
10	746.832	25561.8674	271.532422	28.4035	22
Mean	363.50878	10028.97281	198.0148639	19.38899	15.8

In these five criteria, the two algorithms are capable of competing with each other. Table (6) shows mean value of introduced comparing factors obtained from tables (4 and 5).

Table 7. The mean value of defined criteria

Data	NPS	S	Time	MID	D
NSGA-II	13.1	18.99538	195.1587999	9990.02	244.4582
NRGA	15.8	19.38899	198.0148639	10028.97	363.5088

5.7. Comparison of multi-objective algorithms

In this section, statistical analysis was implemented to compare the two algorithms applying the t test method and Minitab 17 software, using the five criteria.

5.7.1 Maximum spread or diversity (D)

The statistical hypothesis for the equality of the maximum spread or diversity criterion for solved problems is presented in table 8.

$$\begin{Bmatrix} H_0 : \mu_{NSGA-II\,(D)} = \mu_{NRGA(D)} \\ H_1 : \mu_{NSGA-II\,(D)} \neq \mu_{NRGA(D)} \end{Bmatrix}$$

Table 8. T-test for diversity (NSGA-II vs NRGA)

Algorithms	N	Mean	StDev	SE mean
NSGA-II	10	244	249	79
NRGA	10	364	354	112
Difference = μ (NSGA-II) - μ (NRGA)				
Estimate for difference : -119				
95% CI for difference : (-409, 171) T-test of difference=0 (vs ≠) : T-Value = -0.87 P-Value = 0.397 DF = 16				

The output obtained from T-test above, the P-value is 0.397. Since this value is greater than 5%, H_0 is not rejected. In other words, at 95% confidence level, the H_0 is not rejected. The assumption of the equality of the means is not rejected and there is no significant difference between the algorithms of maximum spread or diversity and that both algorithms can compete with each other at this level. The comparison result is depicted in figure (5).

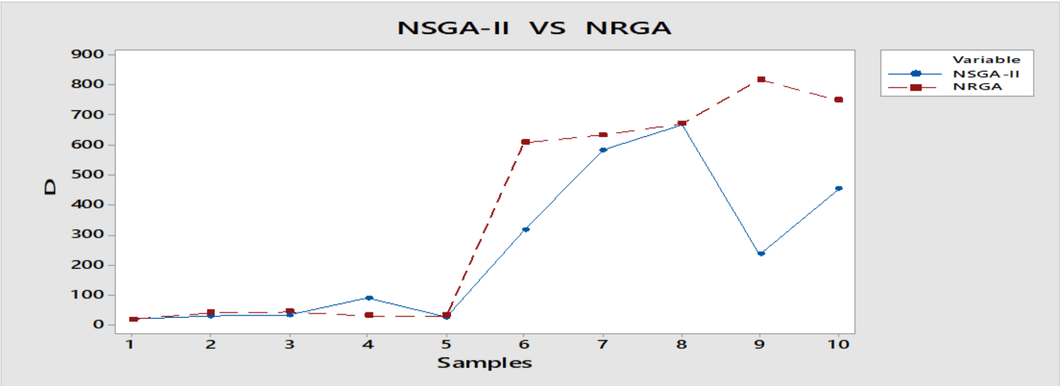


Figure 5. Comparison of the two algorithms based on the maximum spread or diversity (D)

5.7.2. The mean ideal distance (MID)

The statistical hypothesis for the equality of the mean ideal distance criterion for the solved problem is presented in table (9).

$$\begin{Bmatrix} H_0 : \mu_{NSGA-II(MID)} = \mu_{NRGA(MID)} \\ H_1 : \mu_{NSGA-II(MID)} \neq \mu_{NRGA(MID)} \end{Bmatrix}$$

Table 9. T-test for mean ideal distance (NSGA-II vs NRGA)

Algorithms	N	Mean	StDev	SE mean
NSGA-II	10	9990	10323	3264
NRGA	10	10029	10425	3297
Difference = μ (NSGA-II) - μ (NRGA)				
Estimate for difference : -39				
95% CI for difference : (-9827,9749) T-test of difference=0 (vs ≠) : T-Value = -0.01 P-Value = 0.993 DF = 17				

Given the output obtained from T test above, the P-value obtained from the test results is 0.993. Since this value is greater than 5%, the H_0 is not rejected, in other words, at 95% confidence level, the H_0 is not rejected. Therefore, the assumption of the equality of the means is not rejected and there is no significant difference between the algorithms in terms of the mean ideal distance and that both algorithms can compete with each other at this level. The comparison result is depicted in fig(6).

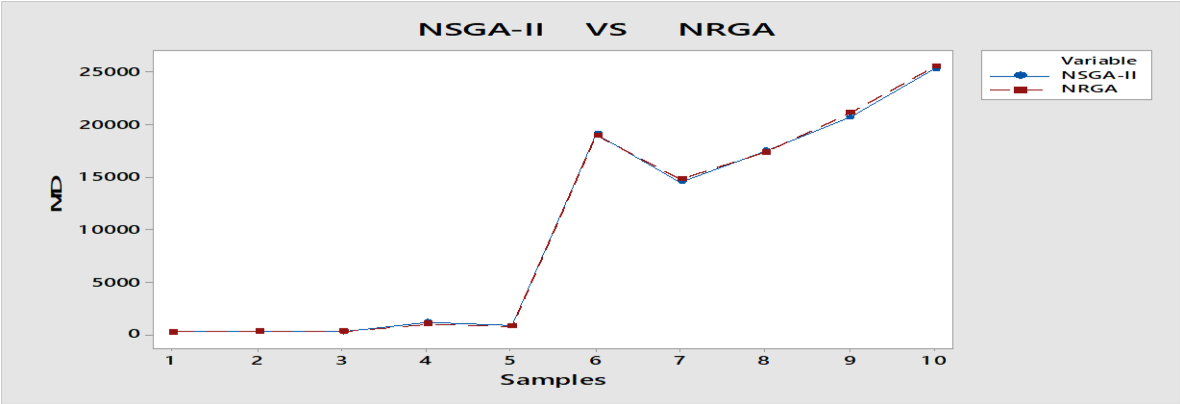


Figure 6. Comparison of the two algorithms based on the mean ideal distance

5.7.3. Time

The statistical hypothesis for the equality of the mean standard solution time for the solved problem is presented in table (10).

$$\begin{Bmatrix} H_0 : \mu_{NSGA-II\,(TIME)} = \mu_{NRGA(TIME)} \\ H_1 : \mu_{NSGA-II\,(TIME)} \neq \mu_{NRGA(TIME)} \end{Bmatrix}$$

Table 10. T-test for mean standard (NSGA-II vs NRGA)

Algorithms	N	Mean	StDev	SE mean
NSGA-II	10	195.20	48.00	15
NRGA	10	198.00	51.60	16.00
Difference = μ (NSGA-II) - μ (NRGA)				
Estimate for difference : -2.9				
95% CI for difference : (-49.9,44.2) T-test of difference=0 (vs ≠) : T-Value = -0.13 P-Value = 0.900 DF = 17				

Given the output obtained from T test above, the P-value obtained from the test results is 0.900. Since this value is greater than 5%, H_0 is not rejected, in other words, at 95% confidence level, H_0 is not rejected. Therefore, the assumption of the equality of the means is not rejected and there is no significant difference between the algorithms in terms of time and that both algorithms can compete with each other at this level. The comparison result is depicted in figure (7).

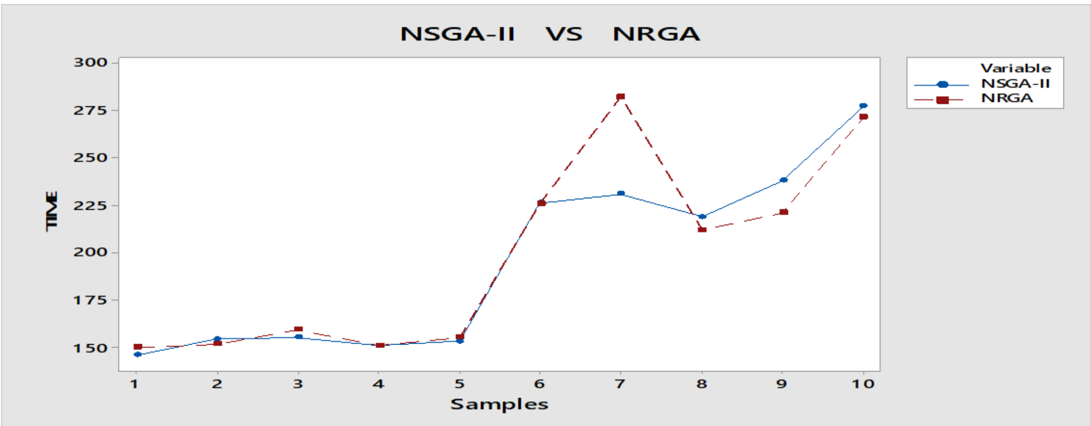


Figure 7. Comparison of two algorithms based on computational time criterion

5.7.4. Spacing (S)

The statistical hypothesis for the equality of the mean spacing criterion for the solved problem is presented in table (11).

$$\begin{cases} H_0 : \mu_{NSGA-II(S)} = \mu_{NRGA(S)} \\ H_1 : \mu_{NSGA-II(S)} \neq \mu_{NRGA(S)} \end{cases}$$

Table 11. T-test for mean spacing (NSGA-II vs NRGA)

Algorithms	N	Mean	StDev	SE mean
NSGA-II	10	19.00	29.80	9.40
NRGA	10	19.40	20.70	6.50
Difference = μ (NSGA-II) - μ (NRGA)				
Estimate for difference : -0.40				
95% CI for difference : (-24.70, 23.90) T-test of difference=0 (vs ≠) : T-Value = -0.0.3 P-Value = 0.973 DF = 16				

Given the output obtained from T test above, P-value obtained from the test results is 0.973. Since this value is greater than 5%, H_0 is not rejected, in other words, at 95% confidence level, H_0 is not rejected. The assumption of the equality of the means is not rejected and there is no significant difference between the algorithms in terms of spacing and that both algorithms can compete with each other at this level. The comparison result is depicted in figure (8).

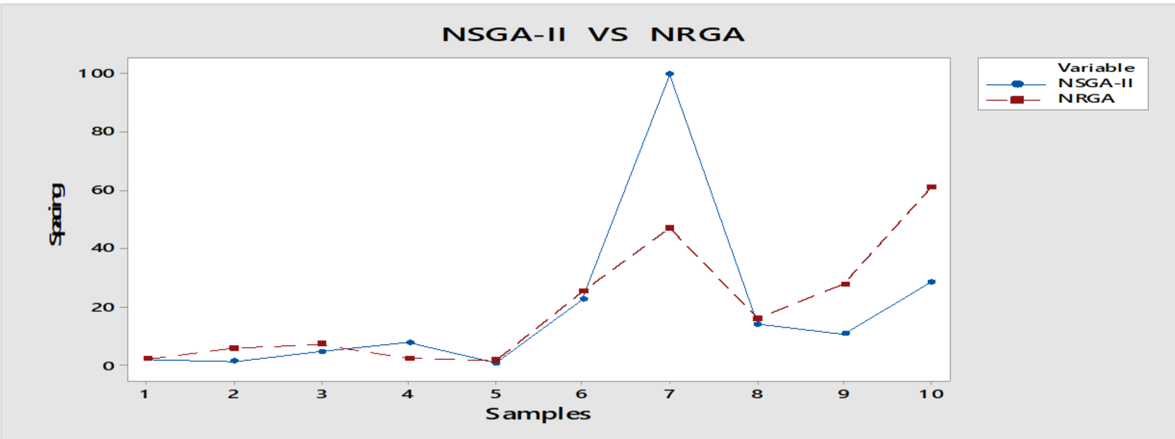


Figure 8. Comparison of two algorithms based on spacing criterion

5.7.4. The number of Pareto solutions (NPS)

The statistical hypothesis for the equality of the number of Pareto solutions for the solved problem is presented in table (12).

$$\begin{cases} H_0 : \mu_{NSGA-II(NPS)} = \mu_{NRGA(NPS)} \\ H_1 : \mu_{NSGA-II(NPS)} \neq \mu_{NRGA(NPS)} \end{cases}$$

Table 12. T-test for number of Pareto solutions (NSGA-II vs NRGA)

Algorithms	N	Mean	StDev	SE mean
NSGA-II	10	13.10	8.89	2.80
NRGA	10	15.80	12.10	3.80
Difference = μ (NSGA-II) - μ (NRGA)				
Estimate for difference : -2.70				
95% CI for difference : (-12.75, 7.35) T-test of difference=0 (vs ≠) : T-Value = -0.57 P-Value = 0.577 DF = 16				

Given the output obtained from T test above, P-value obtained from the test results is 0.577. Since this value is greater than 5%, H_0 is not rejected and at 95% confidence level, H_0 is not rejected. Therefore, the assumption of the equality of the means is not rejected and there is no significant difference between the algorithms in terms of the number of Pareto solutions and that both algorithms can compete with each other at this level. The comparison result is depicted in figure (11).

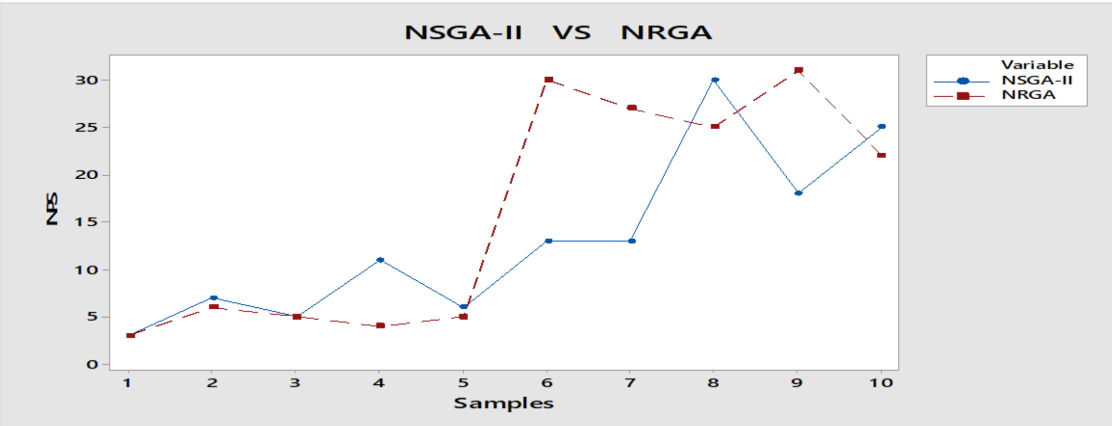


Figure 9. Comparison of two algorithms based on the number of Pareto solutions

5.7.5. Experimental result

Solutions from Pareto front are stored in an elitist list, which is updated each time a new solution from the current population dominates one in the list. The proposed approach differs from NSGA-II in a way it creates a new population. In NSGA-II, the current and offspring population are merged into a new population. Each solution is ranked according to its non-domination level (one is the best level, two is the next-best level) and in case of draw, crowding distance is considered. Next, the binary tournament is used as selection method, but the selection criterion is based on the crowded-comparison operator. In this part, the quality of the results of the two implemented algorithms is evaluated by the five discussed criteria. In this study, the null hypothesis (H_0) is the equivalency of the examined criterion of the solutions of the two algorithms, and the alternative hypothesis (H_1) is the unequal criterion of the solutions. In the above hypothesis, $\mu_1 1$(A) is the average of the criterion A in algorithm 1, and $\mu_1 2$ (A) is the average of the criterion A in the second algorithm. In the following calculations, the t-student test has been used to determine the statistical significance of null hypothesis. The significance level of the test is set to which denotes P-Values with values more than 1% ($\mu / 2$) would reject the null hypothesis. In Figure 10 and 11 set of optimized Pareto solutions and the trend of their improvements in each generation of the two algorithms (NSGA-II and NRGA) are displayed. Overall, we examined the two algorithms for 120 runs (2 algorithms * 12 examples * 5 run per example).

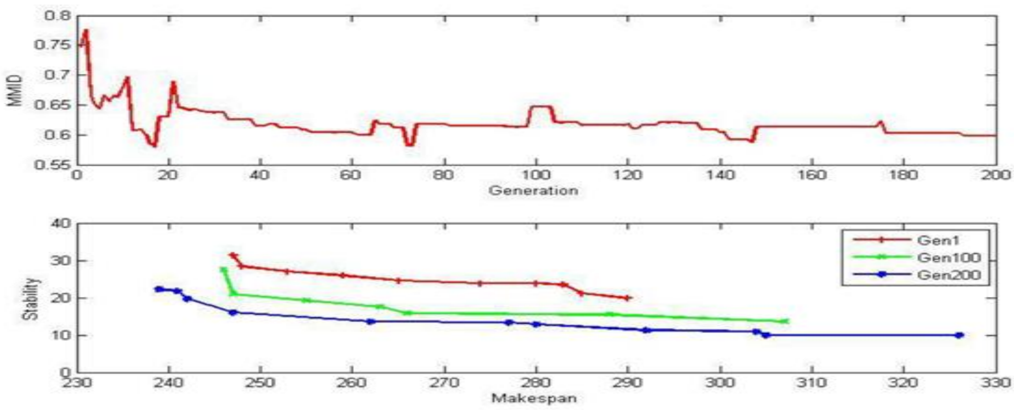


Figure 10. Convergence plot (upper) and Pareto front (lower) of NSGAII for MK10 problem

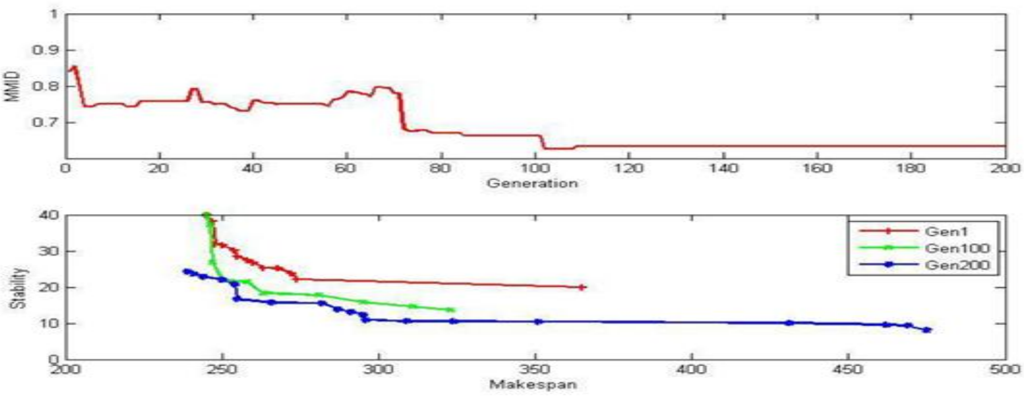


Figure 11. Convergence plot (upper) and Pareto front (lower) of NRGA for MK10 problem

5.8. Decision making using multi-criteria TOPSIS technique for comparing NSGA-II and NRGA algorithms

The Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) model was proposed by Huang and Yun in 1981. This method is one of the techniques used in multi-criteria decision-making (MCDM). In order to compare the combined value of five performance measures, *W*, and CPU time, a popular multi-criteria decision-making method named TOPSIS (Triantaphyllou ,2000) can be used. The considered method ranks the solutions of the problem, which includes conflicting criteria and is helpful for the decision makers to finally decide and make a decision. Table (13-14) illustrates the decision matrix related to the test problems. As two algorithms (NSGAII, NRGA as alternatives) and five performance measures (D, MID, T, S, and NPS as criteria) are considered, the decision matrix consists of two alternatives and five criteria. For employing the TOPSIS method, the decision maker is able to consider the value of the weights according to the criterion's importance. Based on the criterion's importance three modes can occur; (1) *W* and CPU time have the same importance so their weights are equal, (2) *W* is more important for the decision maker so its weight is greater than CPU time, (3) CPU time has more importance than *W* so a greater weight is considered for it. The decision problems are divided into three categories; operational, tactical and strategical problems. As the optimal solution and CPU time, both are important but in the three categorizes mentioned above their importance are somehow different. The optimal solution is more important than CPU time in tactical and strategical problems so it can get a greater weight. However, as the CPU time is more important in operational problems, the considered weight is greater. Since FJSP is considered as tactical and strategical problems, the optimal solution has more importance. For this reason, the second mode is applied which the weights of W and CPU are considered 0.7 and 0.3, respectively. As it can be seen in the second row of Table (13), NSGA-II showed better results compared with NRGA. However, the other two modes are considered and the results are obtained. Table (13) illustrates the results of ranking using TOPSIS method for the three modes mentioned above.

$$W = \sum_{i=1}^{I} wiPi$$

Table 13. Algorithms' comparison results

No. of ex.	NSGA-II		NRGA	
	W	CPU	*W*	CPU
1	7.4584	364.67913	11.6779	308.09,585
2	8.2362	428.56486	12.7768	349.28851
3	8.6949	496.56231	14.7087	246.18175
4	5.6407	719.27633	1.0074	808.78046
5	6.4208	441.26564	8.3333	249.89928
6	7.8493	2184.82376	11.2445	812.08685
7	10.0000	810.18686	14.1046	830.15926
8	14.0116	952.86671	8.2146	2175.73618
9	13.3541	1352.94741	13.4551	1437.83610
10	11.1429	1574.92751	12.3180	1924.74610
11	11.3043	1829.00736	7.0743	2013.73645
12	14.2857	2197.28008	14.9647	1928.15139
Average	9.87	1112.70	10.82	1090.39

In this method, *m* options are evaluated by *n* indices, which must be ranked according to the criteria, or each of them has one performance score. The principle logic of this model defines the ideal solution (positive) and the ideal negative solution. An ideal positive solution is the solution that will increase the profit criterion and reduce the cost criterion. An optimal option is an option that has the least distance from the ideal solution, yet the farthest distance from the ideal negative solution. In terms of rating options in the TOPSIS method, the options with the highest likelihood options achieve a higher rank. In this technique, A^+ and A^- are ideal positive and negative solution, respectively. An A_1 option has a less distance to the ideal positive solution and the more distance to the ideal negative solution than the A_2 option.

5.8.1. The steps of the TOPSIS algorithm

Step 1. The formation of the decision matrix

In the TOPSIS technique, using *n* criterion, *m* options are evaluated. Thus, each option is rated based on a criterion, and these rates can be quantitative or realistic, or qualitative and theoretical. In any case, one decision matrix m * n must be formed.

Step 2. Normalization of the decision matrix

Like other multi-criteria decision-making methods, the decision matrix should be normal and the vector method should be used to normalize the values. In contrast to simple linear normalization, the linear normalization is performed as follows:

$$n_{ij} = \frac{x_{ij}}{\sqrt{\sum_{1}^{m} x_{ij}^2}}$$

Step 3. Generation of a normal weighted decision matrix

The next step is to create a normal matrix based on the weight of the criteria. The criteria of greater importance have higher weights. Each of the weights must be between zero and one and the sum of weights is equal to one. In order to weigh them, the weight of each criterion is multiplied by the values of criterion.
Step 4. Calculation of positive and negative ideals; in this step, for each indicator, positive ideal (A+) and negative ideal (A-) are calculated.
Step 5. Distance from positive and negative ideals and calculation of ideal solution.
In this step, the relative closeness of each option to the ideal solution is calculated. Euclidean distance of each option is calculated from the positive and negative ideal. The final step is to calculate the ideal solution; in this step, the relative closeness of each option to the ideal solution is counted.

$$d_i^+ = \sqrt{\sum_{j=1}^{n} (V_{ij} - V_j^+)^2}$$

$$d_i^- = \sqrt{\sum_{j=1}^{n} (V_{ij} - V_j^-)^2}$$

$$C_i = \frac{d_i^-}{d_i^- + d_i^+}$$

The value of C_i is between zero and one, the closer value to one, the closer solution to the ideal answer and is the better solution.

5.8.2. Implementation of the TOPSIS technique

In this part of the study, we implemented TOPSIS technique to rank two presented algorithms of NSGA-II and NRGA, which results are presented in table (14). So, from the decision maker perspective, the NSGA-II algorithm is chosen. Based on the calculations made and the choice of the NSGA-II, we applied TOPSIS to rank results of NSGA-II for all of presented samples, final ranking for results of sample 8 is presented in table (15) as an example.

Table 14. Ranking algorithms by applying TOPSIS

	di+	di-	ci	result - rank
NSGA-II	0	0.509902	1	1
NRGA	0.509901951	0	0	2

Table 15. Ranking of results for sample 8 after solving by NSGA-II

di+	di-	ci	result – rank
0.047560352	0.145908	0.75417	9
0.036303077	0.170794	0.824705	3
0.139177949	0.068377	0.329439	29
0.164521887	0.067743	0.291663	30
0.077882936	0.16061	0.673437	12
0.061716781	0.126533	0.672155	13
0.046283574	0.159058	0.774602	8
0.084541963	0.105767	0.555765	17
0.119624601	0.085047	0.415528	24

di+	di-	ci	result – rank
0.080030803	0.119126	0.598152	16
0.039143015	0.147991	0.790829	6
0.034110732	0.169334	0.832334	2
0.052434949	0.138248	0.725015	10
0.094576801	0.094985	0.501075	18
0.110901347	0.07518	0.404016	25
0.122480838	0.076672	0.384992	27
0.033654017	0.150717	0.817466	4
0.113065586	0.076383	0.403187	26
0.099548537	0.084683	0.459656	22
0.075224403	0.144258	0.657265	14
0.100554107	0.087551	0.465437	21
0.13633633	0.071982	0.345537	28
0.030462504	0.156057	0.836679	1
0.096772478	0.093611	0.491698	19
0.036786042	0.147498	0.800384	5
0.052256059	0.130916	0.714716	11
0.043118465	0.159406	0.787095	7
0.076962209	0.146532	0.655641	15
0.097031894	0.092659	0.488472	20
0.102643868	0.082041	0.444222	23

Two multi-objective evolutionary algorithms based on the Pareto-NSGA-II and NRGA approach were considered for solving 10 problems. Several different indexes for evaluating the performance of multi-objective algorithms were introduced and algorithms compared in each indicator using statistical methods and T-test separately. Then, TOPSIS method was performed between the mean defined criteria of two algorithms and NSGA-II ranked better and selected. Finally, among the solutions of NSGA-II algorithm, TOPSIS technique was implemented to determine the solutions to each of 10 ranking problems and the appropriate solutions.

6. Managerial insights

Considering both the benefit of managers (cost and erosion) the multi-objective model is developed with strict constraints. In this research, we tried to optimize MO-FJSSP in deterministic condition; managers are free to apply any of the two meta-heuristics algorithms. As the MO-FJSSP is highly complex / NP-Hard and difficult in solving, an effective algorithm is proposed based on NSGA-II and NRGA. The high global convergence of the algorithm is tested by comparing with previous works while its applicability of MO-FJSSP is verified by the calculation of a real-size case. The results reveal that the model is capable to improve the development level scheduling and robustness can provide more useful managerial insights.

7. Conclusion

Scheduling is one of the most important tasks in planning. The scheduling problem consists of two parts, in other words, the solution to the scheduling problem leads to the adoption of two types of decisions: (a) decisions on allocation of resources, and (b) decisions related to the determination of the sequences. The complexity of these problems depends on its environment. One of the issues that arise in scheduling environment in both types of decision-making FJSP. The potential of FJSP problem has caused to be interested by researchers both theoretically and practically. In this research, the researcher first introduced multi-objective mathematical model and then considered two different multi-objective evolutionary algorithms based on the Pareto-NSGA-II and NRGA approach to solve 10 problems for optimization. In addition, the researcher introduced number of indicators to investigate the performance the multi-objective algorithms and compared the algorithms in each index separately. For comparison of the algorithms, statistical methods and T- test were used. After that, TOPSIS method was implemented among the mean criteria of two algorithms. NSGA-II algorithm had a better rank and was selected. Finally, among the solutions of NSGA-II algorithm, TOPSIS technique was used to rank the solutions to each of 10 questions and select the appropriate solution. Based on the results of the model, the results of this research are consistent with the research by Ahmadi et al. (2016).According to findings of this paper; NSGA II had better performance in comparison with NRGA and further the other studies outputs of TOPSIS method confirmed it too.
In order to carry out further research, it is suggested that other researchers examine the FJSP model with existing models in the queue theory in order to control and improve the production process. Considering the random state, enter the tasks into the system, simulate the process and consider the process of processing time and completing tasks as fuzzy or gray in the mathematical model and implement the model.

References

Afrouzy, Alizadeh, Z., Paydar, M.M., and Nasseri, S.H., (2018). "A meta-heuristic approach supported by NSGA-II for the design and plan of supply chain networks considering new product development", *Journal Industrial Engeenring International*,Vol. 14, pp. 95–109.

Ahmadi, E., Zandieh, M., Farrokh, M., and Emami, S.M, (2016). "A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms", *Computers & Operations Research*, Vol. 73,pp. 56-66.

Al Jadan, Omar, Rajamani, Lakishmi., and Rao, R.C., (2008)."Non-dominated ranked genetic algorithm for solving multi-objective optimization problems: NRGA", *Journal of Theoretical and Applied Information Technology*,Vol. 2,No. 3,pp. 269-277.

Al-Hinai, N., and T.Y. El Mekkawy., (2011)."An efficient hybridized genetic algorithm architecture for the flexible job shop scheduling problem", *Flexible Services and Manufacturing Journal*,Vol. 23, No. 1,pp. 64–85.

Brandimarte, P, (1993). "Routing and scheduling in a flexible job shop by tabu search", *Annals of Operations research*,Vol. 41, No. 3, pp. 157-183.

Brucker, P., and Schlie, R., (1990). "Job-shop scheduling with multi-purpose machines", *Computing*, Vol. 45, No. 4,pp. 369-375.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T, (2000). "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II", In: *International Conference on Parallel Problem Solving From Nature*.

Deb, K., Pratap, A., Agrawal, S., and Meyarivan, T., (2002). "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp.182-197.

Frutos, M., Olivera, A. C., and Tohmé, F., (2010). "A memetic algorithm based on a NSGAII scheme for the flexible job-shop scheduling problem", *Annals Of Operations Research*, Vol. 181, No. 1, pp.745-765.

Gambardella, L., and Mastrolilli, M., (1996). "Effective neighborhood functions for the flexible job shop problem", *Journal of scheduling*, Vol. 3, No. 3, pp. 3-20.

Gamila, M.A., and Motavalli, S., (2003). "A modeling technique for loading and scheduling problems in FMS", *Robotics and Computer-Integrated Manufacturing*, Vol. 19, No. 1, pp. 45-54.

Gao, J., Gen, M., Sun, L., and Zhao, X., (2007). "A hybrid of genetic algorithm and bottleneck shifting for multiobjective flexible job shop scheduling problems", *Computers & Industrial Engineering*, Vol. 53, No. 1, pp. 149-162.

Gen, M., and R., Cheng., (2000). *Genetic algorithms and engineering optimization*, Wiley series in engineering design and automation. New York: Wiley.

Goli, A., Tirkoalee, B.E., Soltani, Mehdi, (2019)."A robust just-in-time flow shop scheduling problem with outsourcing option on subcontractors", *Proudction & Manufacturing Research*, Vol. 7, No. 1, pp. 294-315.

Ho, N.B., J.C., Tay, and E. M-K. Lai., (2007). "An effective architecture for learning and evolving flexible job-shop schedules", *European Journal of Operational Research*, Vol. 179, No. 2, pp. 316-333.

Hosseini, Seyed Mohammad Hasan, (2019)."A bi-objective model for the assembly flow shop scheduling problem with sequence dependent setup times and considering energy consumption", *Journal of Industrial Engineering and Management Studies*,Vol. 6, No. 2, pp.44-64.

Jiang, Z., Zuo, L., and Mingcheng, E., (2014)."Study on multi-objective flexible job-shop scheduling problem considering energy consumption", *Journal of Industrial Engineering and Management*, Vol. 7, No. 3, pp. 512-589.

Kacem, I., Hammadi, S., and Borne, P., (2002a). "Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems", *IEEE Transactions on Systems, Man, and Cybernetics, Part C* (Applications and Reviews), Vol. 32, No. 1, pp. 1-13.

Kacem, I., Hammadi, S., and Borne, P., (2002b). "Correction to approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems", *IEEE Transactions on Systems, Man, and Cybernetics, Part C* (Applications and Reviews), Vol. 32, No. 2, pp. 172-172.

Kumar, V. M., Murthy, A., and Chandrashekara, K., (2012). "A hybrid algorithm optimization approach for machine loading problem in flexible manufacturing system", *Journal of Industrial Engineering International*, Vol. 8, No. 1, pp. 13-27.

Lei, D., (2011). "Simplified multi-objective genetic algorithms for stochastic job shop scheduling", *Applied Soft Computing*, Vol. 11, No. 8, pp. 4991-4996.

Li, Yibing., Huang, Weixing., Wu, Rui., and Guo, Kai., (2020)."An improved artificial bee colony algorithm for solving multi-objective low-carbon flexible job shop scheduling problem", *Applied Soft Computing Journal*, Vol. 95, pp. 1-14.

Li, J.-q., and Pan, Q.-k.,(2012). "Chemical-reaction optimization for flexible job-shop scheduling problems with maintenance activity", *Applied Soft Computing*, Vol. 12, No. 9, pp. 2896-2912.

Liu, H., Abraham, A., Choi, O., and Moon, S., (2006). "Variable neighborhood particle swarm optimization for multi-objective flexible job-shop scheduling problems", *Lecture notes in computer science*, 4247, 197.

Moveahed, K., Nama, G., Ali, (2019). "Optimal flexible capacity in Newsboy problem under stochastic demand and lead-time", *Journal of Industrial Engineering and Management Studies*, Vol. 6, No. 1, pp.18-45.

Rahmati, S.H.A., and Zandieh, M., (2012). "A new biogeography-based optimization (BBO) algorithm for the flexible job shop scheduling problem", *The International Journal of Advanced Manufacturing Technology*, Vol. 58, No. 9, pp. 1115-1129.

Shen, Liji, Pers, S., Stephene, Neufled, S., Janies, (2018). "Solving the flexible job shop scheduling problem with sequence -dependent setup times", *European Journal of Operational Research*, Vol. 256, No. 2, pp. 503-516.

Soltani,Mehdi,(2019)." A robust just-in-time flow shop scheduling problem with outsourcing option on subcontractors",proudction & manufacturing research, Vol.7,No.1,pp.294-315.

Tirkoalee, B.E., Goli, A., Weber, Gerhard-Wilhelm, (2019). "Multi-objective aggregate production planning model considering overtime and outsourcing options under fuzzy seasonal demand", *Advances in Manufacturing*, Volume 2 - Production Engineering and Management, pp.81-96.

Triantaphyllou E., (2000). Multi-criteria decision making methods: a comparative study, Kluwer Academic Publishers, Dordrecht.

Tavakkoli-Moghaddam, R., Javadian, N., Khorrami, A., and Gholipour-Kanani, Y., (2010). "Design of a scatter search method for a novel multi-criteria group scheduling problem in a cellular manufacturing system", *Expert Systems with Applications*, Vol. 37, No. 3, pp. 2661-2669.

Xia, W., and Wu, Z, (2005). "An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems", *Computers & Industrial Engineering*, Vol. 48, No. 2, pp. 409-425.

Xiong, J., Xing, L.-n., and Chen, Y.-w., (2013). "Robust scheduling for multi-objective flexible job-shop problems with random machine breakdowns", *International Journal of Production Economics*, Vol. 141, No. 1, pp. 112-126.

Zitzler, E., (1999). Evolutionary algorithms for multiobjective optimization: Methods and applications, Institut für Technische Informatik und Kommunikationsnetze Computer Engineering and Networks Laboratory.

This article can be cited: Ehtesham Rasi, R., (2021). "Optimization of the multi-objective flexible job shop scheduling model by applying NSGAII and NRGA algorithms", *Journal of Industrial Engineering and Management Studies*, Vol. 8, No. 1, pp. 45-71.