RESEARCH PAPER

A developed framework for sequencing of mixed-model assembly line with customer's satisfaction and heterogeneous workers

Masoud Rabbani[1], Seyedeh Zeinab Beladian Behbahan[1], Hamed Farrokhi-Asl[2], Majedeh Esmizadeh[3]

[1]School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran.
[2]School of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran.
[3]School of Business, University of Kansas, Kansas, USA.

ABSTRACT

Goals: We present a multi-objective mathematical model to determine the optimum production sequence of the mixed-model assembly line (MMAL). Maximizing customer satisfaction and minimizing costs are the objectives of the problem.

Design / Methodology / Approach: Customers are divided into two clusters of high priority and low priority by k-medoids method. Also, to get closer to the real world, heterogeneous workers are considered. As the actual scale of the problem cannot be solved by an exact method, two meta-heuristic algorithms, namely Strength Pareto Evolutionary Algorithm 2 (SPEA2) and Non-Dominated Sorting Genetic Algorithm II (NSGA-II) are proposed to solve the problem and reach approximate and efficient results in large scale.

Results: It observes that this model can plan the customers' orders by considering their satisfaction. Also, comparing the results of these algorithms indicates a slight superiority of the SPEA2 method.

Limitations of the investigation: This study is mainly limited by clustering criteria. In the future, more criteria can be considered for analyzing customer behavior and expanding customer clusters.

Practical implications: This model can help all manufacturers who use MMAL by providing a Pareto front for deciding between costs and customers' satisfaction.

Originality / Value: Applying k-medoids to cluster the customers for better orders management and proposing SPEA2 and NSGA-II for solving the problem are the main novelties of this study.

Keywords: Mixed model assembly line; sequencing; SPEA2; customer satisfaction; k-medoids.

INTRODUCTION

A mixed-model assembly line (MMAL) allows different products to be assembled on one line.. A final product may be the result of assembling deferent parts with varying times of processing on a common line. The recent studies of marketing show a tendency for the diversity of products, and in some cases, customers demand different and completely customized products (Ramezanian and Ezzatpanah, 2015). Manufactory with a make-to-order strategy usually starts to produce after their customers choose each feature of products.

Two different producing policies exist in a mixed assembly line: make-to-stock and make-to-order (Manavizadeh et al., 2013). Manufacturers produce orders with the preference of customers in the make-to-order strategy; the production process is started after that

customers select the feature of products that desire. In this strategy, customers' satisfactions depend on the performance of the order's delivery. MMAL with make-to-order can produce continuously and simultaneously different orders in one assembly line (Rabbani et al., 2018b). MMAL is very implemented in industries. Also, variability in the launch interval causes more flexibility in the assembly line.

Most studies of MMAL proposed the same task time and salary for each worker in stations. Therefore, in real situations, workers have a variety of skills. Hence, the processing time depends on the ability of workers and can perform at different speeds. The assumption of Heterogeneous workers impacts the total cost, in which high skill workers require a further salary. Bartholdi and Eisenstein (1996) propose different speeds for workers in the assembly line at the Toyota Sewing.

Customer satisfaction plays an essential role in competitive business environments. Identifying valuable customers to manage customer relationship management (CRM) increases customer satisfaction in which improves market share (Hu et al., 2013; Safari et al., 2016). In this paper, customers are divided into two groups: high priority customer and low priority of the customer. The purpose is to complete the order of high priority customers first, and they get their orders sooner than low priority customers. The division of customer into high and low priority customers is organized with a k-medoids method, which is an efficient method for clustering.

The objectives are maximizing customer satisfaction and minimizing cost included utility worker time and idle time. Two different priority is considered which measure customer satisfaction. Workers have the diverse skill; hence the problem becomes near to the real world. The model is solved and compared with NSGA-II and SPEA2, and the results demonstrate the superiority of SPEA2.

The remainder of this paper is organized as follows. Section 2 reviews the relating literature. Section 3 describes the problem and the model. Section 4 presents the encoding of models. In section 5 compares two algorithms, and section 6 provides the conclusion of the article.

Literature review

MMAL is a kind of assembly line that different products assemble on a single line (Fattahi and Salehi, 2009). Producing different productsat a suitable price is a big challenge for producers. Lots of variety of demands, competitive prices, and making a balance between machines and operators encourage the producer to implement a mixed assembly line. Producer uses this kind of assembly lines for different reasons like more flexibility and the power of answering to different demand of customer (Manavizadeh et al., 2013).

Different objectives are used in the sequencing of MMAL studies. In most studies, the summation of idle costs and utility costs and variable launch intervals are objectives that must become minimized (Fattahi and Salehi, 2009). Manavizadeh et al. (2011) considered production rate variation, utility work, and delay and the earliest cost as objectives. Manavizadeh et al. (2013) proposed six objectives: minimize cost of utility work, total setup cost and cost of variability in rate of production, all of them are used in previous studies and they add three other objectives that are minimizing idle costs, operators' error costs and tardiness costs. In MMAL, concern is about dependent setup time to the sequencing works. Based on this concern, Sungur and Yavuz (2015) focus on the sequencing of orders with minimizing unfinished works for a problem that setup time depends on sequencing. Also, the branch and bound is used for finding the desired sequence, and the heuristic algorithm is used for large-scale problems. Minimization of total cost of utility work, idle work cost and maximization of customers' satisfaction are used as objectives of model of by Rabbani et al. (2018b).

Some studies have been done to investigate the effects of workers' ability and cost factors on the total cost as well as the number of different types of workers assigned to each station. When workers are employed with higher skills, their salaries become more than other

workers. Skill workers increase the speed of doing works, so it can be modeled on how to be employed workers with different skill levels. The results of the study show that increasing cost factors that related to workers will improve their quality and speed. But it also increases costs for a short period. Time factors have similar effects as cost, but cost factors have a greater impact on total costs (Sungur and Yavuz, 2015). Masood Rabbani et al. (2018a) consider MMAL balancing problems with a parallel line in a make-to-order environment. One of the assembly lines utilizes higher skills workers or modern technology. Hence, it causes more cost but increases the efficiency of the line. It is assumed to minimize operation cost, cycle time, and smoothness index.

There are many aspects which consider measuring customer satisfaction. It is necessary to handle and select data from the customer to know what influences on satisfaction and dissatisfaction of customers. This measurement makes manufacturers understand how to organize services to provide the highest value of comfort and also inform manufacturers which factors are under the control of them (de Arruda Falcão et al., 2017). One of the methods for evaluating customers is using recency, frequency, and monetary (RFM), which identify customers that make more profit for the company (Hu and Yeh, 2014). Recency (R) informs about how customers desire their purchases, and it is the time between the last consumption behavior and the current date. Frequency (F) refers to the number of transactions in a given period, and the monetary (M) refers to the amount of money which spend in a particular period (Coussement et al., 2014). Peker et al. (2017) extended this method by adding two new criteria, namely Length (L) and Periodicity (P). With this method, customers can be prioritized, and according to these priorities, they determine the planning and sequencing of tasks. Rabbani et al. (2018b) use RFM and customers' satisfaction and divide orders into two categorized of high priority orders and standard priority orders. Logically after prioritizing, the focus is on customers with high priority because a high percentage of a company's profit depends on them. After prioritizing customers' orders, a mathematical model is used to determine the order of products on the assembly line.

Data analyzes are used commonly in modern scientific research, like communication science, computer and biology science. Clustering plays an important role as the main factor of data analyzes. Multiple tools have been developed for cluster analyses. Each clustering algorithm has its own strengths and weaknesses (Xu and Tian, 2015). The main idea of clustering algorithms is considering the data center as the cluster center. The most well-known clustering algorithms are k-means and k-medoids. The notion of k-means is to find k cluster centers. It works iteratively to cluster data according to data features, and it updates the center points in each iteration and repeats calculations until the convergence criterion is met. This algorithm allocates each object to the nearest cluster center. The center of each cluster is the median coordinate of points in the cluster. k-means is very efficient in terms of computational time, but it is sensitive to data outside of the range. For these reasons, k-medoids is developed. K-medoids is improved of k-means since this algorithm is based on a centralized point in a cluster, then it is less sensitive to instability than k-means (Park and Jun 2009). In this algorithm, some data points are considered to be the center of the cluster, which is called medoids. Therefore, the k-medoids algorithm is a medoids displacement algorithm.

Many meta-heuristic algorithms are developed for solving multi-objective models which calculate near optimized solution in a reasonable time, for example, Albana et al. (2014) focus on the application of meta-heuristic algorithms in the MMAL problem. They surveyed three algorithms: Genetic Algorithm (GA), Simulated Annealing (SA) and hybrid method based on both Genetic Algorithm and Simulated Annealing (GASA). The results are shown that GA and GASA are suitable methods for industrial users because they are fast enough and make better solutions than manual solutions. Ramezanian and Ezzatpanah (2015) compare performance ICA with GA on the MMAL balancing problem. Also, Zandieh and Moradi (2017) examined the performance of ICA with SA and GA. The results of both studies represent the superiority of ICA to the rest of the algorithms. Zhang et al. (2018) studied the mixed-model assembly line car sequencing problem. They proposed a hybrid algorithm that consists of Tabu search, Large

Neighborhood Search (TLNS), Parallel Constructive Heuristic (PCH), and the Small Neighborhood Search (SNS). The comparison of two algorithms NSGA-II and Multi-objective with Particle Swarm Optimization (MOPSO) is shown by Rabbani et al. (2018a). They concluded that NSGA-II is more efficient in the sequencing problem. According to these critical advantages of meta-heuristic algorithms, in this study SPEA2 is proposed and its performance is compared with NSGA-II.

Based on the literature review, it can be seen that there is no suitable method for analyzing customers' behavior in customer management and sequence planning in the MMALs. Therefore, in this study, we try to examine the behavior of customers by considering the LRFMP model and divide them into high and low priority clusters with the k-medoids algorithm. Because of the importance of high priority customers, maximizing their satisfaction is considered as one of the objectives of the sequencing of MMALs. In addition, minimizing idle cost and utility cost are regarded as another objective. Since, in reality, the ability of workers varies from one other, in this study, we examine the workers as heterogeneous. One cannot overlook the fact that workers have the ability to learn, which speeds up their work overtime. Since a few studies are considered workers' learning in their models, this criterion is also discussed in this study. Accordingly, the innovation of this study can be summarized as follows:

- Customer clustering into two categories with high and low priority based on the LRFMP model using the k-medoids algorithm
- Workers are considered heterogeneous; also, they are able to learn over time.
- Consider the dependencies of setup times on the latest on-line model and the model that will entire to the line
- Development of two meta-heuristic algorithms namely NSGA-II and SPEA2 for solving the sequencing orders and comparing their results

PROBLEM DESCRIPTION

In this section, two important works must have done: clustering and sequencing. When orders come, they should be clustered into two groups of high priority and low priority according to their features that determine with LRFMP. After that, the sequence of assembly line must be determined. In Figure1, the steps are shown.
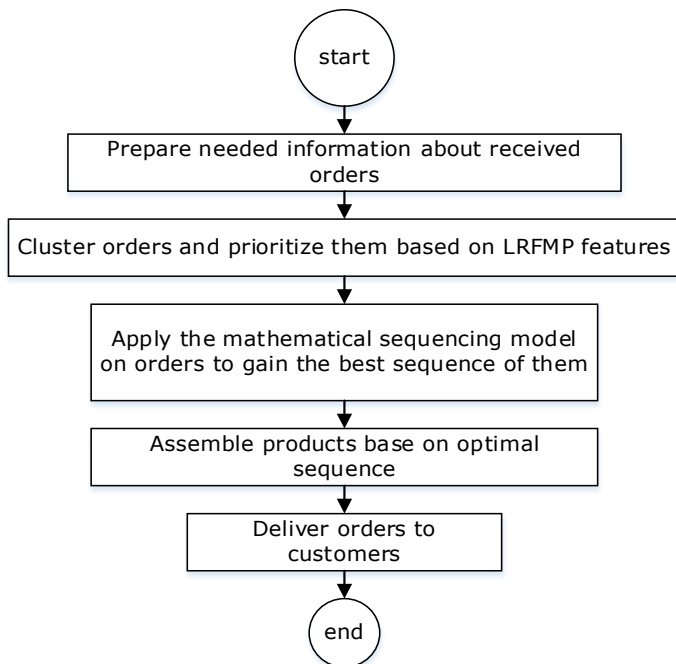
Figure 1. Schematic of the proposed framework. Source: The authors themselves.

Prioritizing orders based on LFRMP

In this section, a k- medoids algorithm is applied to the orders to cluster them into high and low priority orders. The k-medoids method is one of the well-known and efficient algorithms for clustering. It is like a k-means algorithm but has advantages over the k-means. For example, k-means does not have a sound output when there are some outlier data, and while -medoids is not sensitive to the outlier data and because of this, it's a reasonable choice. For dividing n object to k cluster, it chooses k data randomly which is called 'medoids.' Medoids is a member of the cluster that has the minimum dissimilarity to all of the members of the cluster (Xu and Tian, 2015; Sheng and Liu, 2006). In this study, customers divided into two groups. The steps of the algorithm are shown in Figure 2. Two important points must be considered in using the k-medoids method. First, an appreciative distance measurement must be selected, in which Euclidean distance is used in this study. Second, appreciate criteria must be defined for measuring the distance between two-point which Length, Recency, Frequency, Monetary, and Periodicity (LRFMP) is used here.
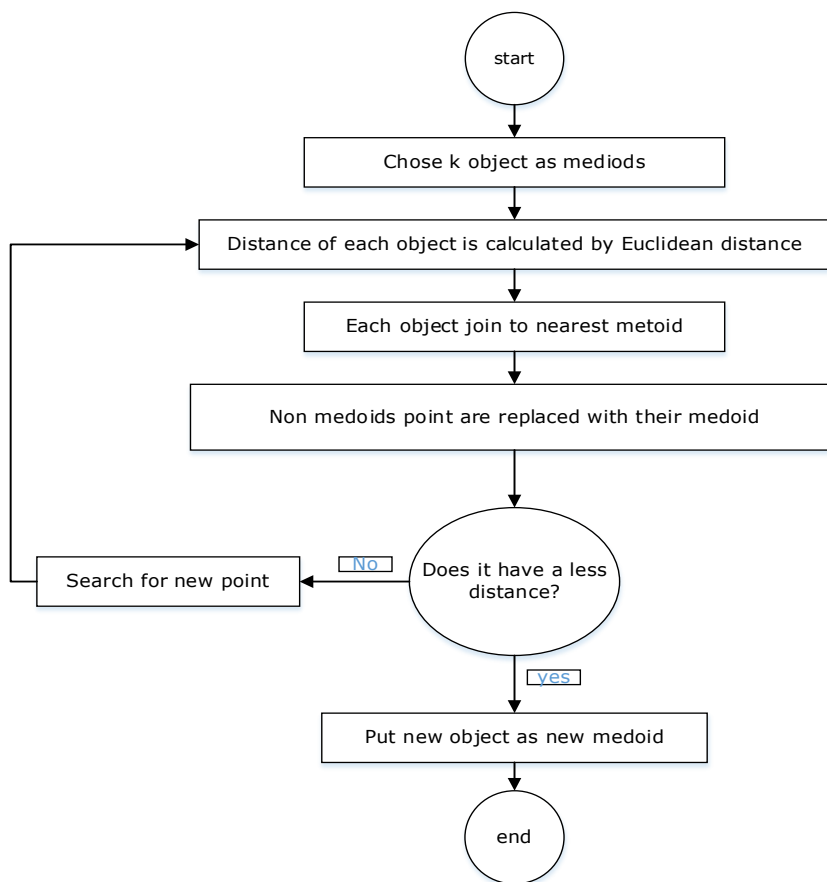

Figure 2. K-medoids algorithm. Source: The authors themselves.

LRFMP is extended to RFM, which is a well-known method for analyzing customer behavior and their value. Peker et al. (2017) define LRFMP as below:

Length(L): is the time interval between the first and last order of a customer. How high this criterion be, the loyalty of the customer is more.

Recency(R): informs about how customers desire for their purchases. Recency is formulated as (Equation 1):

$$R = \frac{1}{m} \sum_{k=1}^{m} z d \left(d_e, d_{n-k+1} \right) \tag{1}$$

It calculates the average of $zd(d_e, d_{n-k+1})$ which shows the difference between the last day of visiting and the day that customer K visits recently. And whatever it becomes lower, the customer is more valuable.

Frequency(F): denotes the number of customers visit and the higher it is, customers are more valuable and loyal.

Monetary(M): denotes the money which is spent during a period of time by customers.

Periodicity(P). It presents how regular customers make an order, and it is calculated with a standard deviation of observation time of customers by equations 2 and 3:

$$P = SD(ob_1, ob_2, ..., ob_m) \tag{2}$$

$$ob_1 = zd(d_{t+1}, d_k) \tag{3}$$

Which *ob* presents observation time and *m* shows customer's number of observation time and d_k indicates the day of the *i*th visit of customers.

This algorithm does not give the value and importance degree of each cluster and just clusters the data. Then, for understanding the value of each cluster and mark them as a high or low priority group, the Analytic hierarchy process (AHP) is implemented. To apply AHP, acquired data must have been normalized using Equation 4. Then, using AHP, the weight of each cluster is obtained, and by having these weights, the lifetime value of each cluster is determined using Equation 5. For more information about AHP, interested readers can refer to Vargas (1990). Finally, the cluster with low CLV and high CLV are determined as low priority and high clusters.

$$y_i = \frac{x_i}{\underset{i=1}{\overset{N}{Max}}(x_i)} \tag{4}$$

$$CLV(Cluster_i) = W_L L + W_R R + W_F F + W_M M + W_P P \tag{5}$$

Sequencing assumption

Customers' orders have a different priority, and they are divided into two groups of high priority and low priority. Customer satisfaction is clustered based on LRFMP factors. This satisfaction depends on the time of delivery of the order to customers. Using fuzzy function, it is possible to show the satisfaction level of the customer according to the delivery time of their orders (Rabbani et al., 2018b).

$$Customer\ satisfaction = \begin{cases} 0 & Delivery\ time \leq ee \\ \dfrac{Delivery\ time - ee}{E - ee} & ee \leq Delivery\ time \leq E \\ 1 & E \leq Delivery\ time \leq L \\ \dfrac{ll - Delivery\ time}{ll - L} & L \leq Delivery\ time \leq ll \\ 0 & ll \leq Delivery\ time \end{cases} \tag{6}$$

Where *ll* and *ee* denote the maximum and minimum time that is acceptable for a customer that the orders achieve to them. *E* and *L* show earliness and lateness time that the order must receive to customers.

This study is assumed a conveyor belt with constant speed *S* and have S closed-type workstation. Different products are come to line with rate *f*. In each station, workers move downstream with the workpiece. If a worker cannot finish the work of his station, utility

workers complete the unfinished work. And workers back to the start point of the workstation (Rabbani et al., 2018b). It is considered a vector $p = (d_1,...,d_n)$, which n shows the whole type of models, and d_n denotes the number of orders of model n. The launch interval is calculated as below with Equation 7:

$$\theta = \frac{total\ time\ that\ requires\ for\ all\ vector\ p}{o\sum_{n=1}^{N} d_n} \tag{7}$$

Other assumptions that are used in this study are as follow:
- Workers have different skill, so it makes a limitation on time and finance
- Allocation of each worker to each workstation because of different skill that they have
- Customer's orders depend on their value which has different priority
- Destruction of machines do not survey in the study
- All of the orders received to the customer at the same time
- Each customer can have different orders
- Earliness and lateness time are considered for each order that are determined by the customer
- *ee* and *ll* are defined by experts.
- the transfer time of worker between the first and end of his station is not considered
- processing times are deterministic

Also,two objectives are discussed in this study, maximizing customer satisfaction and minimizing total cost.

Sequencing mathematical model

To determine the best sequence of orders based on these objectives, the mathematical model of Rabbani et al. (2018b) is developed. Also, the notations of this model are below (see the Notation Chart 1). And the model is presented as Equations 8 to 35.

$$min f_1(y) = \sum_{j \in J} \sum_{o \in O} (Ati_{oj} + Btu_{jo}) \tag{8}$$

$$max f_2(y) = \sum_{i \in I} (\alpha H_i + 1) x_{2,i} \left(\frac{tc_i - ee}{E - ee} \right) + \sum_{i \in I} (\alpha H_i + 1) x_{3,i}$$
$$+ \sum_{i \in I} (\alpha H_i + 1) x_{4,i} \left(\frac{ll - tc_i}{ll - L} \right) \tag{9}$$

$$tc_j = \theta(j-1) + max \left\{ \sum_{n} \sum_{o} J''_{o,n} y^2_{n,j} , \sum_{o} \frac{k_o}{s} \right\}$$

$$\forall j \tag{10}$$

$$tc_i \le ee + M(1 - x_{1,i}) \quad \forall i \tag{11}$$

$$ee - M(1 - x_{2,i}) < tc_i \le E + M(1 - x_{2,i}) \quad \forall i \tag{12}$$

$$E - M(1 - x_{3,i}) < tc_i \le L + M(1 - x_{3,i}) \quad \forall i \tag{13}$$

$$L - M(1 - x_{4,i}) < tc_i \le ll + M(1 - x_{4,i}) \quad \forall i \tag{14}$$

Chart 1. Notation Chart.

Sets	Definition
O	Set of the station; $o = \{1,\ldots,O\}$
$n, n^\bullet$	Set of different kind of models; $n = \{1,\ldots,N\}$
I	Set of orders; $i = \{1,\ldots,I\}$
J	Set of positions; $j = \{1,\ldots,J\}$
W	Set of workers; $w = \{1,\ldots,W\}$
Parameters	
E	earliness time that the orders must receive to customers
L	lateness time that the order must receive to customers
ee	the minimum time that is acceptable for the customer that the orders achieve to them
ll	the maximum time that is acceptable for the customer that the orders achieve to them
$pt_{n,n'}^{o}$	setup time in station o if the model n' becomes after model n
S	speed of a conveyor belt
θ	launch interval
α	level of importance for high priority orders
$q_{i,n}$	quantity of order i for model n
k_o	length of station o
t_{no}^{w}	Normal processing time of model n at station o for worker w
A	cost of an idle worker per time
B	cost of utility worker per time
H_i	1 if order i is high priority order; 0 otherwise
β_o	Rate of workers learning coefficient in station o
Decision variables	
$x_{1,i}$	1 if order i arrives before ee; otherwise 0
$x_{2,i}$	1 if order i arrives in the interval $[ee, E]$; otherwise 0
$x_{3,i}$	1 if order i is arrived in the interval $[E, L]$; otherwise 0
$x_{4,i}$	1 if order i arrives in the interval $[L, ll]$; otherwise 0
$x_{5,i}$	1 if order i arrives after ll; otherwise 0
tc_i	Completion time of order i
ts_j	Time of completion of a product in jth sequence place
ti_{oj}	Idle time for the product in jth sequence place at oth station
tu_{jo}	Utility worker time in jth sequence place at oth station
$t'_{o,n}$	Processing time for model n in station o after worker assignment
$t''_{o,n}$	Processing time for model n in station o which consider workers learning rate
$y_{i,j}^{1}$	1 if a copy of the product for the ith order in sequence j; 0 otherwise
$y_{n,j}^{2}$	1 if a copy of the product for the model n in sequence j; 0 otherwise
fp_{oj}	Starting position at station o on the jth product
WO_{o}^{w}	1 if worker w is assigned to oth station; 0 otherwise

$$ll < tc_i + M\left(1 - x_{5,i}\right) \quad \forall i \tag{15}$$

$$\sum_{k=1}^{5} x_{ki} = 1 \quad \forall i \tag{16}$$

$$tc_i = \max_j \left\{ y_{i,j}^1 ts_j \right\} \quad \forall i \tag{17}$$

$$\sum_i y_{i,j}^1 = 1 \quad \forall j \tag{18}$$

$$\sum_j y_{n,j}^2 = \sum_i q_{i,n} \quad \forall n \tag{19}$$

$$\sum_j y_{i,j}^1 = \sum_n q_{i,n} \quad \forall i \tag{20}$$

$$\sum_O WO_o^w = 1 \quad \forall w \tag{21}$$

$$\sum_w WO_o^w = 1 \quad \forall o \tag{22}$$

$$t_{o,n}' = \sum_w WO_o^w t_{no}^w \quad \forall o,w \tag{23}$$

$$t_{o,n}'' = t_{o,n}' i^{-\beta_o} \quad \forall o,n,i \tag{24}$$

$$\sum_n y_{n,j}^2 = 1 \quad \forall j \tag{25}$$

$$fp_{1,o+1} = \sum_{o'=1}^{o} k_{o'} \quad \forall o = 1,\ldots,o-1 \tag{26}$$

$$fp_{j+1,o} = fp_{j,o} + S\left\{ \sum_n y_{n,j}^2 \left(t_{o,n}'' + \sum_{n'} y_{n',j-1}^2 pt_{n,n'}^o \right) - tu_{j,o} - \theta + ti_{j+1,o} \right\}$$

$$\forall o, j=1,\ldots,j\text{-}1 \tag{27}$$

$$tu_{j,o} \geq \frac{fp_{j,o} + S\left\{ \sum_n y_{n,j}^2 \left(t_{o,n}'' + \sum_{n'} y_{n',j-1}^2 pt_{n,n'}^o \right) \right\} - \sum_{o'=1}^{o} k_{o'}}{S}$$

$$\forall o, j=1,\ldots,j\text{-}1 \tag{28}$$

$$tu_{J,o} \geq \frac{fp_{J,o} + S\left\{ \sum_n y_{n,j}^2 \left(t_{o,n}'' + \sum_{n'} y_{n',j-1}^2 pt_{n,n'} \right) - \left(\sum_{o'=1}^{o} k_{o'} + S\theta \right) \right\}}{S}$$

$$\forall o \tag{29}$$

$$di_{o,j} \geq \frac{\sum_{o'=1}^{o-1}k_{o'} - \{fp_{j-1,o} + S(\sum_n y_{n,j-1}^2(t''_{o,n} + \sum_n y_{n',j-2}^2 pt^o_{n,n'}) - tu_{j-1,o} - \theta)\}}{S}$$

$\forall o, j=2,...,j\text{-}1$ (30)

$$tc_i, ts_j, tu_{jo}, ti_{oj}, fp_{oj} \geq 0 \quad \forall i, j, o$$ (31)

$$t'_{o,n}, t''_{o,n} \geq 0 \quad \forall n, o$$ (32)

$$wo_o^w \in \{0, 1\} \quad \forall w, o$$ (33)

$$x_{1,i}, x_{2,i}, x_{3,i}, x_{4,i}, x_{5,i} \in \{0, 1\} \quad \forall i$$ (34)

$$y_{i,j}^1, y_{n,j}^2 \in \{0, 1\} \quad \forall i, j, n$$ (35)

Utility and idle costs and customer satisfaction are considered as two different objectives functions. Equation 8 shows minimization of costs for idle and utility worker time. Equation 9 maximizes satisfaction of customers. It is formulated using Equation 6. Constraint (10) calculates the total completion time for each product. Constraints (11) - (16) are used for the second objective to figure in which stage the satisfaction happens. They determine the range of completion time of each order. Constraint (17) calculates the delivery time for each order. Constraint (18) ensures that just one product for *i*th order is in sequence *j*. constraint (19) is shown the amount of each model for each order, and (20) denotes the quantity of each order. Constraints (21) and (22) ensure that each station is just related to one worker, and each worker assigns to only one station. If worker *w* is assigned to *o*th station, constraint (23) calculates each processing time. Constraint (24) calculates processing time after adding worker learning rate. Constraint (25) ensures that only one product with model *n* is in sequence *j*. Equation 26 declares that the starting position of the first product equal to the length of all stations that products pass from them. Equation 27 denotes the location of *j*th product that depends on the previous product. Equation 28 and 29 calculate utility worker's time in each station. Equation 30 calculates the idle time for an order of each sequence and each station. Equations 31-35 define the binary and not negative decision variables.
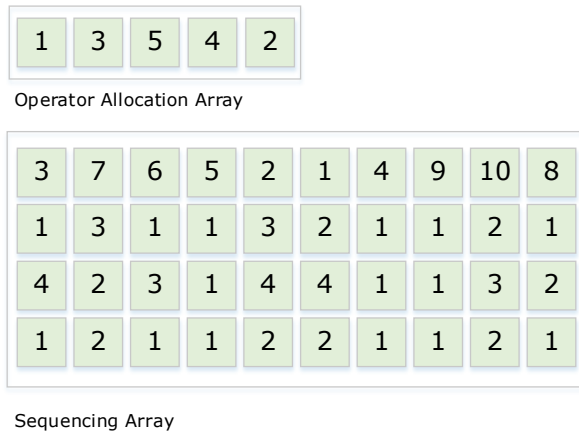
METHODOLOGY

The proposed model is a non-linear mathematical model, and also, it is categorized as an NP-hard problem. That means it is hard to gain an optimal solution for a large-scale problem. We developed NSGA-II and SPEA2 to solve this model. For using meta-heuristic algorithms, three main steps must be done. First, the problem must be encoded. Then according to developed encoding, the objectives must be calculated that are known as fitness in these algorithms. Finally, according to the procedure of algorithms, solutions must be improved. In the next part, the encoding of the model and method of NSGA-II and SPEA are explained.

Encoding the problem

The encoding's way of problem influences on performance of algorithm. Encoding of the problem must include all feasible solutions and solution space. One of the well-known and operational methods is encoding by sorting (Chen et al., 2013). In this study, each chromosome solution of the problem has two arrays in their structures. The first array has one row that each cell declares one station and number in each cell show several workers that

assigns to this cell. The second array relates to the sequence of orders that includes four rows. The first row indicates the orders' number; the second one suggests customer number which order belongs to him, the third one indicates the model of each order, and the fourth row indicates clustering of customers. Also, columns of this array show the sequence of orders. By having an allocation of workers and sequencing of orders, it's now possible to compute the objectives. For example, Figure 3 is shown a schematic for a problem that has five stations, ten orders, and three customers with four different types of models.

1	3	5	4	2

Operator Allocation Array

3	7	6	5	2	1	4	9	10	8
1	3	1	1	3	2	1	1	2	1
4	2	3	1	4	4	1	1	3	2
1	2	1	1	2	2	1	1	2	1

Sequencing Array

Figure 3. An example of encoding of problem. Source: The authors themselves.

Proposed NSGA-II

NSGA-II is the famous meta-heuristic algorithm for discrete problems (Srinivas and Deb, 1994). This algorithm starts its work by generating random solutions set, which are named population. Then using its operators, absolute numbers of children are created in each iteration, and fitness of each solution is calculated. In this step, it merges the children with the population and makes the new population. Then with non-dominated sorting method keeps a certain number of better solution and remove the others.

This algorithm iterates and produces the new children until the end of the iteration criterion reaches. Most of the time, the stop criterion is some iterations, which is used here too. Finally, this algorithm returns a set of approximate Pareto solutions. The pseudo-code of this algorithm is shown in Figure 4.

```
Set Algorithm parameters : Size of population (Np), Maximum number of iteration, percent of
crossover children, percent of mutation children

Create initial population randomly

Sort the population based on none dominated sorting method

For i=1:Maximum number of iteration

      Create Crossover children

      Create Mutation children

      Merge children and population

      Sort the new population based on none dominated sorting
      method

       Keep Np members of the best population's members

End

Return the population
```

**Figure 4.** Pseudo code of NSGA-II. Source: The authors themselves.

To find solutions that are close to the optimal solution, it is necessary to scan the entire solution space. This algorithm generates two kinds of children to improve the solutions and scape from the local optimality. The first set of children, which are created by combining two parents are crossover children whose purpose is to improve the solutions. In addition, the second set of children born of one parent is named mutation children whose mission is to escape from local optimality. Selecting of parents is random, and the way that new child is produced is as below:

- **Generating of crossover child:** Based on the chromosome structure, which has two arrays, a single point crossover operator is applied to each of these arrays. This operator selects a random number from one to n, which n is the length of the array. Then according to the chosen number, the parents divide into two parts. In the following, the first part of the first parent with the second part of the second parent is combined and creates a new child and the first part of the second parent with the second part of the first parent combined and create another child. An example of creating children for sequencings array in this way is shown in Figure 5.

Parent 1 / Child 1 / Parent 2 / Child 2
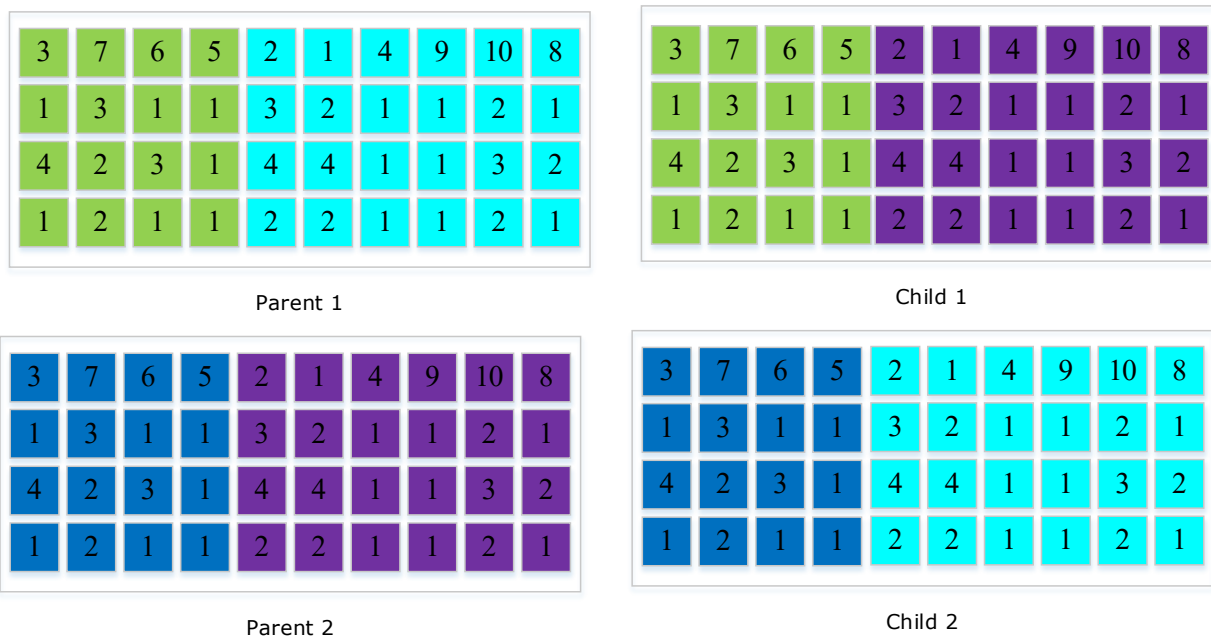
Figure 5. An example of crossover operator on the sequencing array. Source: The authors themselves.

- **Generating of mutation child:** This operator takes two cells of each array of selected parents and changes these two with each other. An example of this operator for the first array is presented in Figure 6.

Parent / Child
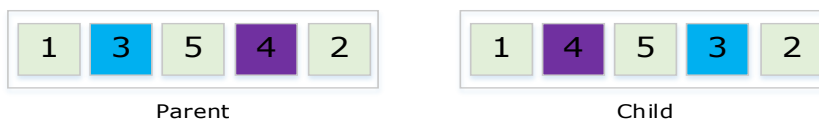
Figure 6. An example of mutation operator on the assignment array. Source: The authors themselves.

Proposed SPEA2

SPEA2 is developed based on SPEA by Zitzler et al. (2001). First, it creates an initial random population, then non-dominated members are moved to a solution archive. After, it calculates fitness for an initial solution and solution archive. If members of solution archive are more than certain members, extra members are removed from the archive using clustering methods, which for clustering, k-medoids can be used. Then the tournament binary selects some members from initial and archive members. New generation are made using crossover

and mutation operators. This algorithm is optimized for combination of suitable fitness determination's approaches, density estimation techniques, and advanced archive size reduction methods. Interested readers can refer to Zitzler et al. (2001) for more information on this algorithm. The pseudo code of this algorithm is shown in Figure 7.

Set algorithm parameters : Size of population ($N(P)$), Size of achieve $N(A)$, Maximun number of iteration, Percent of crossover children, Percent of mutation children

Create initial population randomly and set Archive $\emptyset$

Sort the population based on none dominated sorting method

For i=1:*Max_iter*

    For $Si \in$ Population

        Si objectives ← CalculateObjectives(Si);

    End

    Union ← Population + Archive;

    For $Si \in$ Union

        Si raw ← CalculateRawFitness(Si , Union);

        Si density ← CalculateSolutionDensity(Si , Union);

        Si fitness ← Si raw + Si density ;

    End

    Archive ← GetNonDominated(Union);

    if Size(Archive) < $N(A)$

        PopulateWithRemainingBest(Union, Archive, $N(A)$);

    else if Size(Archive) >$N(A)$

        RemoveMostSimilar(Archive, $N(A)$);

    End

    Selected ← SelectParents(Archive, $N(P)$);

    Population ← CrossoverAndMutation(Selected, P crossover ,P mutation);

End

Return the NonDominated Archive

Figure 7. Pseudo code of SPEA2. Source: The authors themselves.

RESULT AND DISCUSSION

In this section, the first two proposed algorithms are evaluated and compared with each other. Then, an instance is solved, and sensitivity analysis is done on its essential parameters.

Parameter tuning

The parameters of meta-heuristic algorithms must be tunned to enhance their performance. So for tuning the parameters of proposed algorithms, Taguchi method, which is a great method for the design of the experiment is used. Four parameters, including a maximum of iteration (max_iter), number of population (npop), percent of crossover (pc), and percent of mutation (pm) are effective on the performance of NSGA-II. Also, the performance

of SPEA2 depends on the maximum of iteration (max_iter), many the archive (narchive), percent of crossover (pc), and percent of mutation (pm). Using Minitab software, A three-level Taguchi experiment is designed for finding the optimal value of parameters of each algorithm. These experiments are done on a small-scale instance of the problem. In the end, outcomes are shown in Figures 8 and 9, and the optimal value of parameters is in Table 1.
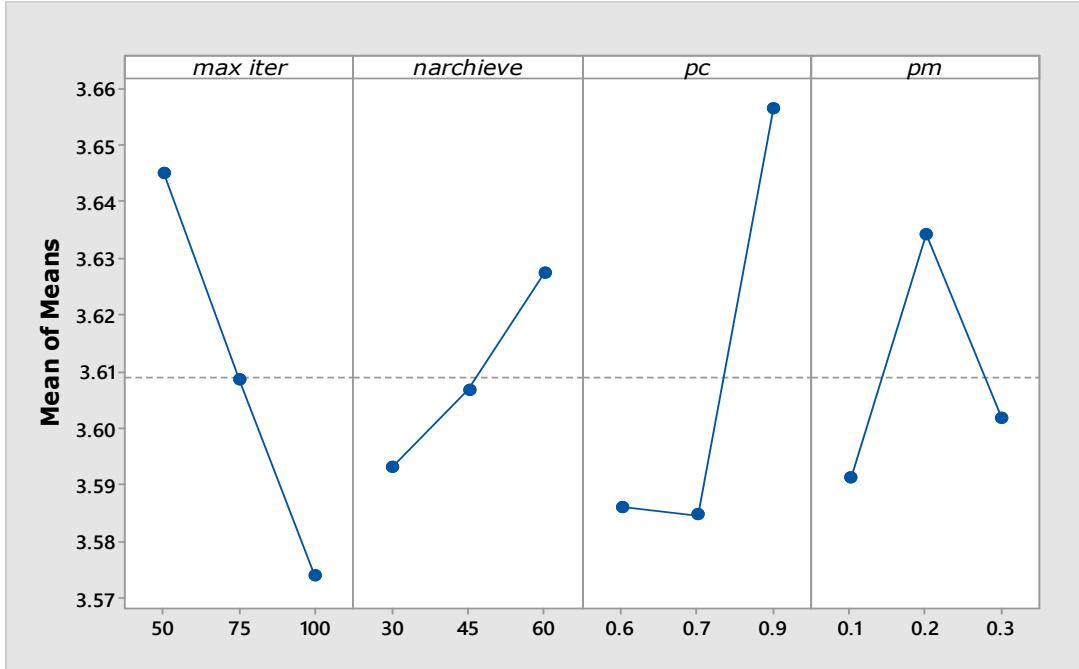

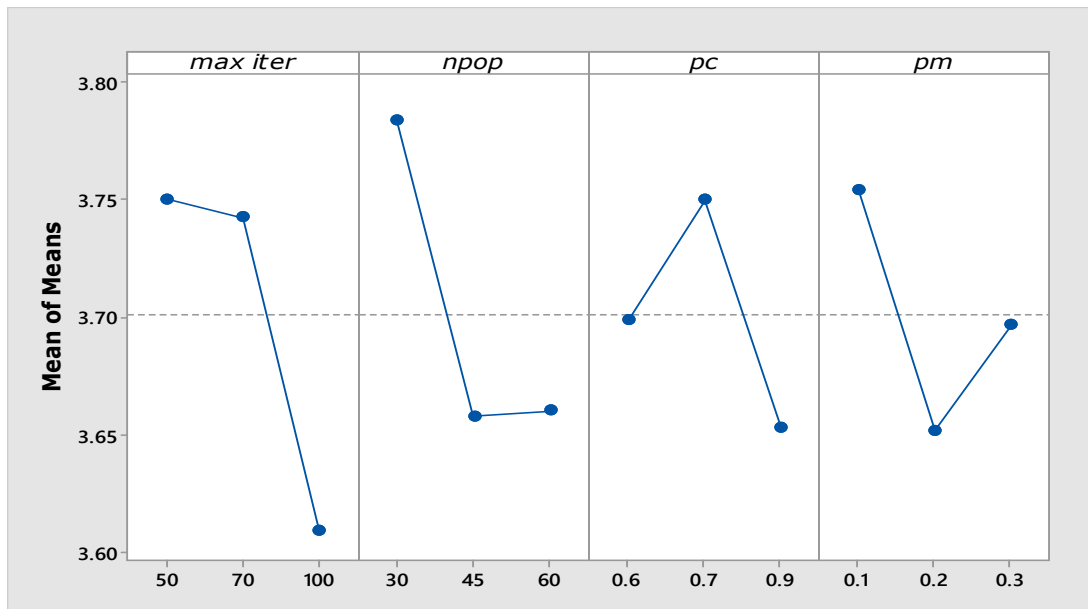**Figure 8.** Analysis diagram for SPEA2. Source: The authors themselves.

Figure 9. Analysis diagram for NSGA-II. Source: The authors themselves.

Table 1. Best value of proposed algorithm parameters

Algorithm	Max-iter	Npop	n-archive	Pc	Pm
SPEA2	100	-	30	0.7	0.1
NSGA-II	100	45	-	0.9	0.2

Source: The authors themselves.

Evaluating the performance of NSGA-II and SPEA

For evaluating and comparing the performance of proposed algorithms, in this section, 10 different instances with different characteristics are solved and survived. These instances create with a random instance code creator in Matlab. The main features of these instances are in Table 2. Parameters are produced randomly with Matlab, and the ranges of parameters are in Table 3. Also, the results of the comparison of the two algorithms are in Table 4. For comparing the quality of returned solutions of each algorithm and investigate the performance of them, four criteria are used, which are mentioned as follows:

Runtime: This criterion shows the runtime of each algorithm.

The number of Pareto solutions: this criterion depicts the number of optimal Pareto solutions. Algorithm performance is better if the value of this criterion is higher.

Spacing: this criterion measures with Equation 36 and 37. In these equations, N_i indicates the non-dominated solution, $\bar{N}$ is the mean of N_i and m indicates the number of solutions. y_k^i is an optimal solution for all $k \in \{1,...,m\}$

$$Spacing = \sqrt{\frac{1}{J-1} \sum_{j=1}^{J} (N_i - \bar{N})^2}$$

(36)

$$N_j = \min_{i=1}^{I} \left(\sum_{k=1}^{K} \left| y_k^i - y_k^j \right| \right)$$

(37)

Diversity: This criterion denotes how an algorithm distributes in objectives space and measures as the maximum Euclidean distance between non-dominated solutions, which as more it is, algorithm performance is better.

Comparing based on run time, a number of Pareto solution, spacing, and diversity are in Figures 10-13, respectively.

Table 2. Main features of test instances

No.	Based characteristics			
	Number of customer	Number of model	Number of station	Total orders
1	15	3	10	76
2	20	3	10	93
3	30	3	10	140
4	30	3	15	130
5	30	4	15	178
6	35	4	15	232
7	40	4	15	257
8	40	5	15	302
9	50	5	15	384
10	60	5	15	454

Source: The authors themselves.

Table 3. Range of producing random parameters

Parameter	Lower bound	Upper bound
ee	600	800
E	$ee+600$	$ee+800$
L	$E+600$	$E+800$
ll	$L+400$	$L+800$
$pt_{n,n'}^o$	0	100

Table 3. Continued...

Parameter	Lower bound	Upper bound
$q_{i,n}$	0	3
k_o	60	65
t_{no}^{w}	35	60
A	10	20
B	20	30
β_o	0	1

Source: The authors themselves.

Table 4. Proposed algorithms' comparison results

NO.	Number of Pareto solution		Runtime		Spacing		Diversity	
	SPEA2	NSGA-II	SPEA2	NSGA-II	SPEA2	NSGA-II	SPEA2	NSGA-II
1	1	1	1122	976	0	0	0	0
2	1	1	1153	987	0	0	0	0
3	1	3	1211	1018	0	1521	88	5587
4	8	10	1212	1046	94	232	9158	22862
5	6	6	1252	1081	500	1024	52395	23936
6	9	5	1314	1126	28	544	33718	24983
7	11	16	1339	1184	25	561	7711	166310
8	10	13	1345	1188	274	45	39450	59670
9	7	9	1425	1239	187	62	11613	22924
10	5	2	1502	1317	33	0	12696	3318
Average	6	6.6	1287	1116	114	399	16683	32959

Source: The authors themselves.
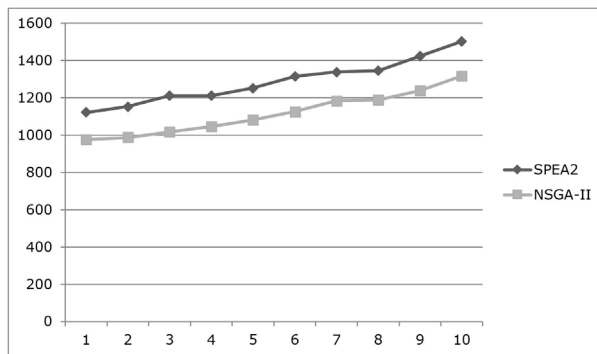
Figure 10. Comparison of algorithms base on run time. Source: The authors themselves.
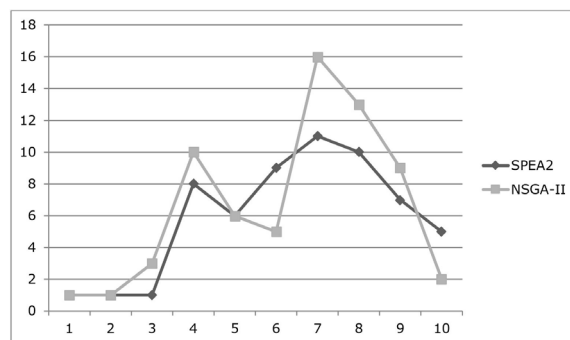
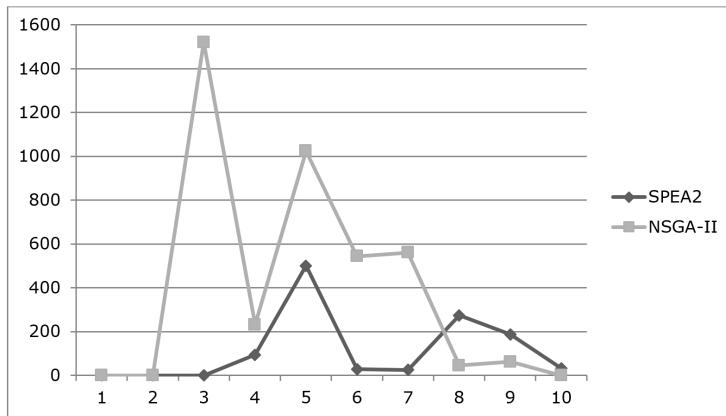**Figure 11.** Comparison of algorithms based on Number of Pareto solution. Source: The authors themselves.

Figure 12. Comparison of algorithms base on spacing. Source: The authors themselves.
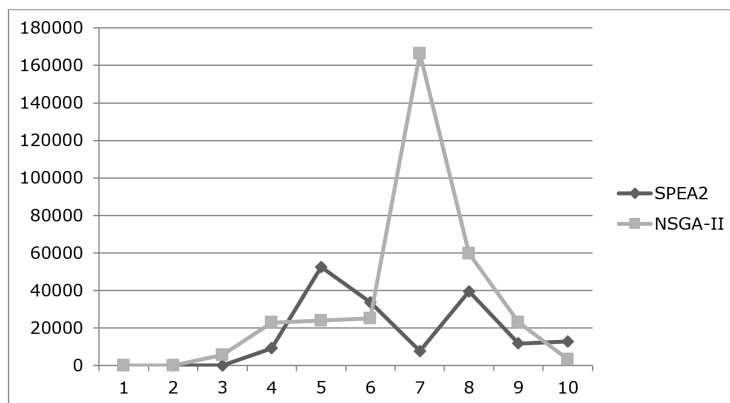

Figure 13. Comparison of algorithms base on diversity. Source: The authors themselves.

According to the results of the experiments shown in Figures 10-13, it can be seen that the performance of these two algorithms is not significantly different from each other based on the mentioned criteria. In terms of the runtime, performance of NSGA-II is slightly better. There is no difference in terms of the number of Pareto solutions. In terms of spacing, the performance of SPEA2 is better, and finally, they do not differ significantly in diversity.

Though according to the proposed criteria, these two algorithms have the same function, what can be a significant difference in comparison between algorithms performance is the solutions provided by them. For example, for instances 5 and 6, these solutions are shown in Figures 14 and 15.
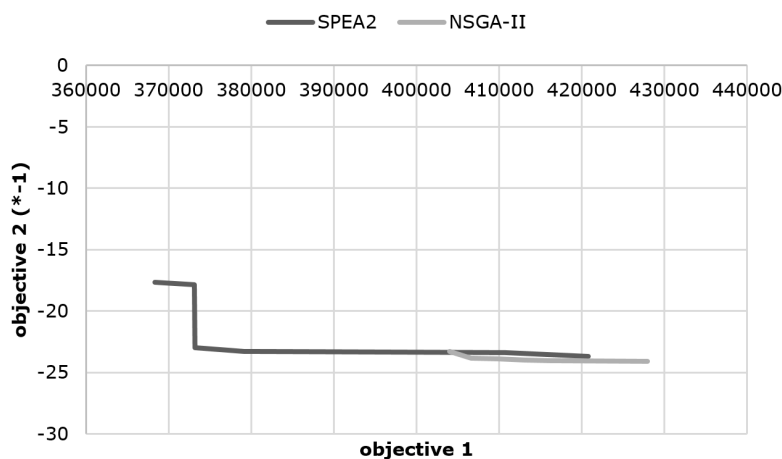

Figure 14. Pareto front for instance 5. Source: The authors themselves.
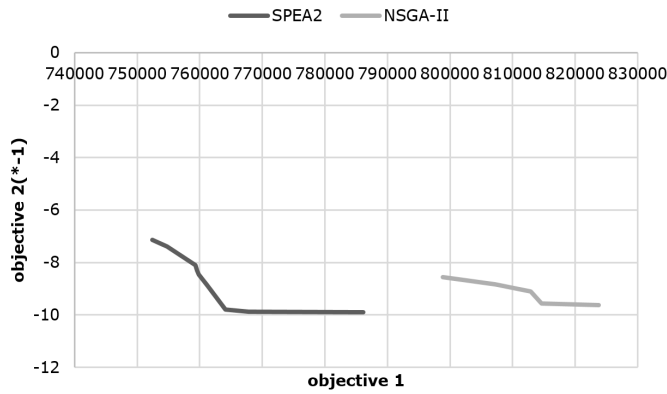
Figure 15. Pareto front for instance 6. Source: The authors themselves.

Regarding the comparison of the solutions provided by the two algorithms in the experiments, which two of them are shown in Figures 14 and 15, it is observed that the SPEA2 offers more optimize solutions than NSGA-II. In most cases, NSGA-II's solutions are overcome by SPEA2's solution, which indicates the superiority of SPEA2. Ultimately, according to the criteria and experiments, it can be concluded that SPEA2 is preferred to NSGA-II, and its usage is more appropriate for solving MMAL sequencing problems.

Sensitivity analysis

The two critical parameters that affect the model are setup time ($pt_{n,n'}^{o}$) and learning rate of workers (β_o). For this reason, the sensitivity of the objective functions against these parameters is analyzed in this section. For this purpose, the instance 7 has been selected. The changes in the objective functions against the learning rate are shown in Figure 16 and its changes against the setup times are shown in Figure 17.
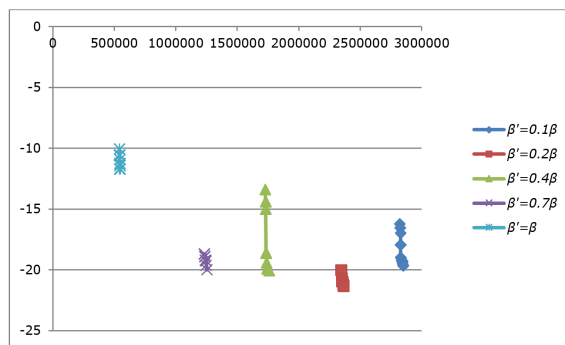

Figure 16. Sensitivity analysis of β_o. Source: The authors themselves.
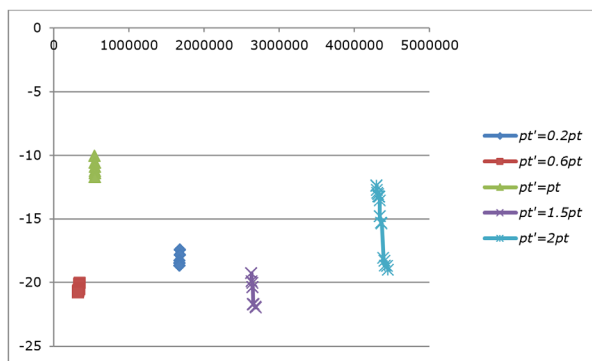

Figure 17. Sensitivity analysis of $pt_{n,n'}^{o}$. Source: The authors themselves.

As it can be seen in Figure 16, changes in the learning rate affect the costs and significantly customer satisfaction. These changes are such that the costs are reduced with increasing learning rates. Also, in Figure 17, it can be seen that changes in setup time also have a small impact on customer satisfaction; their significant impacts are on costs.

CONCLUSION

In this study, a multi-objective problem for the sequencing of MMALs is presented with maximizing customer satisfaction and minimizing idle cost and utility cost. To pay more attention to valuable customers, they are divided into two clusters with high priority and low priority; then the customer satisfaction is just calculated for high priority ones. Customers' clusters are based on LRFMP criteria. The k-medoids algorithm is also used to classify the customers; one of the most essential advantages of this algorithm is the lack of sensitivity to outlier data. In order to get closer to the real world, heterogeneous workers are considered in the sequencing model, which affects the speed and time of the work process. While sequencing problem is NP-hard and there is no exact method for it in large scale, SPEA2 and NSGA-II algorithms have been used to solve it. The meta-heuristic algorithms have their own specific parameters that affect their performance and should be tuned, in which the Taguchi method is implemented for this purpose. The parameters that are given of the two algorithms are almost identical and the results of parameter tuning are shown that these parameters are approximately the same for both algorithms. Then, in the next section, these algorithms are generally compared based on four criteria, the number of Pareto solution, runtime, spacing, and diversity, besides the provided solutions by these algorithms. As a result, the SPEA2 algorithm has better performance than NSGA-II. Using these results, experts who work on assembling lines can choose a better algorithm to achieve better performance.

All in all, this study can provide a framework for companies making a different kinds of commodities in the assembly line scheme and manifests the importance of the learning of workers on the performance of all of the system and customer satisfaction level. What is more, in today's complex world, companies are facing some different and somewhat conflicting objectives. The algorithmic approach provided in this paper so as to trade-off between conflicting objectives can be lucrative tools for assisting individuals who are facing with such condition in real world problems.

In future research, more items can be added to the cost function, such as tardiness and earliness costs. Also, more criteria can be considered for analyzing customer behavior and expanding customer clusters. And other meta-heuristic algorithms can be implemented and compared with SPEA2, to select the most appreciate algorithms.

REFERENCES

Albana, A.S., Aroui, K., Alpan, G. et al. (2014), "Mixed Model Assembly Line Sequencing to minimize delays using meta-heuristics" in *Joint International Symposium IMSS'14 and CIE'44*. Istanbul, Turkey.

Bartholdi 3rd, J.J. and Eisenstein, D.D. (1996), "A production line that balances itself", *Operations Research*, Vol. 44, No. 1, pp. 21-34. http://dx.doi.org/10.1287/opre.44.1.21.

Chen, C.F., Wu, M.C. and Lin, K.H. (2013), "Effect of solution representations on Tabu search in scheduling applications", *Computers & Operations Research*, Vol. 40, No. 12, pp. 2817-25. http://dx.doi.org/10.1016/j.cor.2013.06.003.

Coussement, K., Van den Bossche, F.A. and De Bock, K.W. (2014), "Data accuracy's impact on segmentation performance: Benchmarking RFM analysis, logistic regression, and decision trees", *Journal of Business Research*, Vol. 67, No. 1, pp. 2751-8. http://dx.doi.org/10.1016/j.jbusres.2012.09.024.

de Arruda Falcão, L.M.A., de Barros Jerônimo, T., de Melo, F.J.C. et al. (2017), "Using the SERVQUAL model to assesssmall service quality and customer satisfaction", *Brazilian Journal of Operations & Production Management*, Vol. 14, No. 1, pp. 82-8. http://dx.doi.org/10.14488/BJOPM.2017.v14.n1.a9.

Fattahi, P. and Salehi, M. (2009), "Sequencing the mixed-model assembly line to minimize the total utility and idle costs with variable launching interval", *International Journal of Advanced Manufacturing Technology*, Vol. 45, No. 9-10, pp. 987-98. http://dx.doi.org/10.1007/s00170-009-2020-0.

Hu, Y.H., Huang, T.C.K. and Kao, Y.H. (2013), "Knowledge discovery of weighted RFM sequential patterns from customer sequence databases", *Journal of Systems and Software*, Vol. 86, No. 3, pp. 779-88. http://dx.doi.org/10.1016/j.jss.2012.11.016.

Hu, Y.H. and Yeh, T.W. (2014), "Discovering valuable frequent patterns based on RFM analysis without customer identification information", *Knowledge-Based Systems*, Vol. 61, pp. 76-88. http://dx.doi.org/10.1016/j.knosys.2014.02.009.

Manavizadeh, N., Dehghani, A. and Rabbani, M. (2011), "Mixed model assembly line sequencing in make to order system with available to promise consideration", *World Academy of Science, Engineering and Technology*, Vol. 57, pp. 707-8.

Manavizadeh, N., Tavakoli, L., Rabbani, M. et al. (2013), "A multi-objective mixed-model assembly line sequencing problem in order to minimize total costs in a Make-To-Order environment, considering order priority", *Journal of Manufacturing Systems*, Vol. 32, No. 1, pp. 124-37. http://dx.doi.org/10.1016/j.jmsy.2012.09.001.

Park, H.S. and Jun, C.H. (2009), "A simple and fast algorithm for K-medoids clustering", *Expert Systems with Applications*, Vol. 36, No. 2, pp. 3336-41. http://dx.doi.org/10.1016/j.eswa.2008.01.039.

Peker, S., Kocyigit, A. and Eren, P.E. (2017), "LRFMP model for customer segmentation in the grocery retail industry: a case study", *Marketing Intelligence & Planning*, Vol. 35, No. 4, pp. 544-59. http://dx.doi.org/10.1108/MIP-11-2016-0210.

Rabbani, M., Alipour, F., Farrokhi-Asl, H. et al. (2018a), "Using metaheuristic algorithms for solving a mixed model assembly line balancing problem considering express parallel line and learning effect", *Brazilian Journal of Operations & Production Management*, Vol. 15, No. 2, pp. 254-69. http://dx.doi.org/10.14488/BJOPM.2018.v15.n2.a8.

Rabbani, M., Heidari, R. and Farrokhi-Asl, H. (2018b), "A bi-objective mixed-model assembly line sequencing problem considering customer satisfaction and customer buying behaviour", *Engineering Optimization*, Vol. 50, No. 12, pp. 1-20. http://dx.doi.org/10.1080/0305215X.2018.1431234.

Ramezanian, R. and Ezzatpanah, A. (2015), "Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem", *Computers & Industrial Engineering*, Vol. 87, pp. 74-80. http://dx.doi.org/10.1016/j.cie.2015.04.017.

Safari, F., Safari, N. and Montazer, G.A. (2016), "Customer lifetime value determination based on RFM model", *Marketing Intelligence & Planning*, Vol. 34, No. 4, pp. 446-61. http://dx.doi.org/10.1108/MIP-03-2015-0060.

Sheng, W. and Liu, X. (2006), "A genetic k-medoids clustering algorithm", *Journal of Heuristics*, Vol. 12, No. 6, pp. 447-66. http://dx.doi.org/10.1007/s10732-006-7284-z.

Srinivas, N. and Deb, K. (1994), "Muiltiobjective optimization using nondominated sorting in genetic algorithms", *Evolutionary Computation*, Vol. 2, No. 3, pp. 221-48. http://dx.doi.org/10.1162/evco.1994.2.3.221.

Sungur, B. and Yavuz, Y. (2015), "Assembly line balancing with hierarchical worker assignment", *Journal of Manufacturing Systems*, Vol. 37, pp. 290-8. http://dx.doi.org/10.1016/j.jmsy.2014.08.004.

Vargas, L.G. (1990), "An overview of the analytic hierarchy process and its applications", *European Journal of Operational Research*, Vol. 48, No. 1, pp. 2-8. http://dx.doi.org/10.1016/0377-2217(90)90056-H.

Xu, D. and Tian, Y. (2015), "A comprehensive survey of clustering algorithms", *Annals of Data Science*, Vol. 2, No. 2, pp. 165-93. http://dx.doi.org/10.1007/s40745-015-0040-1.

Zandieh, M. and Moradi, H., (2017). An imperialist competitive algorithm in mixed-model assembly line sequencing problem to minimise unfinished works. *International Journal of Systems Science: Operations & Logistics*, Vol. 6, No. 2, pp. 1-14.

Zhang, X.Y., Gao, L., Wen, L. et al. (2018), "A hybrid algorithm based on tabu search and large neighbourhood search for car sequencing problem", *Journal of Central South University*, Vol. 25, No. 2, pp. 315-30. http://dx.doi.org/10.1007/s11771-018-3739-2.

Zitzler, E., Laumanns, M. and Thiele, L., 2001. *SPEA2: Improving the strength Pareto evolutionary algorithm*, TIK-report No. 103, Swiss Federal Institute of Technology (ETH), Zurich.