Annals of Optimization With Applications

www.anowa.reapress.com

Ann. Optim. Appl. Vol. 1, No. 2 (2025) 13–27.

Paper Type: Original Article

# Model by Considering Customers' Order Priority and Minimizing Machine Providing an Operation Scheduling Configuration Time

**Aysan Moheb[1,*], Ali Donyavi[1]**

[1] Department of Industrial Engineering, Urmia University, Urmia, Iran; moheb.aysan@yahoo.com; a.doniavi@urmia.ac.ir.

**Citation:**

| | |
|---|---|
| *Received: 3 October 2024*<br>*Revised: 13 December 2024*<br>*Accepted: 15 February 2024* | Moheb, A., & Donyavi, A. (2025). Model by considering customers' order priority and minimizing machine providing an operation scheduling configuration time. *Annals of optimization with applications*, *1* (2), 13-27. |

## Abstract

Production scheduling is one of the issues in the field of planning in production systems that has a significant impact on reducing costs, increasing productivity, customer satisfaction, and other competitive advantages. Orders are received instantly based on the planned schedule, and all customers expect to receive their orders as soon as possible. In a situation where the volume of orders is high, and there are limited product manufacturers, manufacturing companies tend to prioritize their customers. The purpose of this study is to prioritize the customers of a medical equipment company, and customers are prioritized based on some criteria such as deprivation, urgency of orders, purchase volume, good pay, and participation in a regional exhibition. In this research, a two-objective production flow scheduling problem is presented. The first objective function is related to minimizing the weighted delay at the time of delivery. The second objective function is related to minimizing the total program time changes of the devices or setting them. Due to the complexity of the problem, a multi-objective particle swarm optimization algorithm is proposed to solve the problem. In order to evaluate the efficiency of the proposed model, the one-month orders of the manufacturer in question have been examined, and the stated results show the efficiency of the scheduling.

**Keywords:** Flexible job shop production scheduling, Multi-objective particle swarm optimization, Prioritization, Configuration time.

## 1 | Introduction

Today, due to the complexity of manufacturing industries and the need for more efficiency, shorter production cycles, more flexibility, better product quality, more customer satisfaction, and, at the same time, lower costs, the face of production has changed. Today's organizations, to survive, not only have to adapt to this changing business environment, but to survive in such a market, they must think about creating a

Corresponding Author: moheb.aysan@yahoo.com

competitive advantage in such a situation. To achieve such benefits, organizations are looking for ways to optimize operations in their production systems. Production scheduling is one of the most critical issues in production systems. Process planning and production scheduling are two of the most important production functions that are performed separately and in sequence.

These two functions are interdependent, and their integration is essential for the optimal use of production resources [1]. Production scheduling plays a key role in many production systems; an effective production plan is a vital factor that enables the industry to improve production efficiency and optimize resource utilization [2].

In production scheduling, the allocation of available production resources to tasks and decision-making in the sequence of operations are taken into account, and the objectives of the problem are optimized by considering the existing constraints [3]. One of the most popular production scheduling issues is the job shop production schedule, which is NP-hard [4]. In the case of job shop production scheduling, a set of tasks is processed on a limited set of machines.

According to its production routine, each task is processed on machines with a certain processing time, and each machine can only process one operation for each task [5]. In flexible job shop production scheduling, there is a set of machines that are selected for each operation, which increases the flexibility and complexity of scheduling [2]. Hence, flexible job shop production scheduling is more complex than job shop production scheduling. Since flexible job shop production scheduling has many applications in a set of real-world issues, it has received much attention [5].

In the present study, the integration of process planning and scheduling in a flexible job shop manufacturing system with multiple objective functions has been studied and modeled. The research model is presented in a medical equipment company. Customer orders in this model are placed in production planning as soon as they are received. Two objective functions are considered for the model. The first objective function is to minimize the weighted sum of delays in-order delivery, and the second objective function is to minimize the total device set-up times.

In order to calculate the delay times, customers are prioritized, and based on the weight assigned to them, delivery delay calculations are performed. To solve the problem, the particle swarm optimization algorithm is used, and the results of the problem indicate its efficiency. The structure of the present study is as follows: The second section provides an overview of previous research in the field of process planning and scheduling. In the third part, the research model and the proposed algorithm for solving the research are explained. In the fourth section, the results of model solving are given. In the last section, the summary of the research and suggestions for future research are mentioned.

## 2 | Literature Review

The issue of flexible job shop scheduling is one of the most critical issues in the production of various smithereens, which is common in systems with low production volume, wide variety, and orders for manufacturing [6]. Among the various scheduling models, the scheduling of flexible job shop production is of great importance. The flexible scheduling issue deals with scheduling a set of tasks, each task involving one or more operations, and each operation can be performed on a set of machines [7]. Flexible job shop scheduling is divided into two groups.

The general, flexible job shop scheduling allows all devices to process all different task operations in the job shop, and the partial flexible job shop scheduling, in which only various task operations can be processed by some of the machines in the job shop. Solving methods for the flexible job shop production scheduling issue are divided into three categories: Precise, innovative, and meta-innovative algorithms. Accurate algorithms are usually modeled and solved by integer linear programming or complex integer linear programming models. In these studies, in which innovative methods are used to solve them, it is common to compare innovative methods with accurate complex integer programming in terms of minimizing workload, balancing work sequences in flexible production systems, and presenting a simulation model to evaluate the performance of the system. Population-based meta-innovative methods are widely used to solve flexible job shop production scheduling [8].

Özgüven et al. [9] developed a Mixed-Integer Linear Programming Model (MILP-1) for the Flexible Job Shop Scheduling Production Model (FJSPs) and compared it with an alternative model (Model F) to confirm its superiority. Besides, they modified MILP-1 in order to lead to MILP-2. Özgüven et al. [10] also formulated two complex integer ideal modeling models for the flexible job shop production scheduling problem, which cover the flexibility of the process program and the separability or inseparability of the start-up sequence. In the first model, the sequence time is inseparable from the startup, and in the second model, it is separable.

Jahromi & Tavakkoli-Moghaddam [11] presented a new integer linear programming model (1-0) considering the dynamic issue of machine tool selection and operations allocation with the tool and part movement policies in flexible production systems. This model aims to determine the composition of machine tools for each piece operation by minimizing production costs. Li et al. [6] presented a mathematical model of the flexible job shop production scheduling problem. A hybrid algorithm of a bee colony with Tabu search was proposed to solve the model. Comparison of the results with the published algorithms and the results of the analysis show that the proposed algorithm was efficient.

Shen et al. [12] presented a flexible job shop production problem in which the three objectives of time interval, maximum machine workload, and stability to uncertainty are simultaneously considered under a variety of practical constraints. A modified multi-objective evolutionary algorithm for sustainable planning was also proposed [12]. Gaeo et al. [8] solved the problem of flexible job shop production planning by learning the rules of dispatch using random forest.

Shen et al. [13] modeled the problem of flexible multi-objective (Combined) job shop production planning, which is widely used in real production systems. They proposed an optimal Improved Non-Dominated Sorting Biogeography-Based Optimization (INSBBO) algorithm to solve the problem.

Homayouni et al. [14] presented an FJSPT, which can be considered as the development of an FJSP and Job Shop Scheduling Production Transport model (JSPT). The optimal solutions of this model were obtained by using the Biased Random Key Genetic Algorithm (BRKGA) genetic algorithm and combining it with a greedy innovative solution to select the processing machine of each operation and the vehicles transporting the processes.

Özgüven et al. [10] considered the production schedule of a flexible job shop in a sheet metal processing company. The goal was to produce a model and an algorithm to make a weekly production plan for the company. The study aimed to minimize production time while meeting the demand for products for a planned horizon. Then, to solve the genetic algorithm, the Giffler and Thompson algorithm and three innovative algorithms were developed [15].

# 3 | Research Model

The scheduling of N product operations by M machines will be examined in this research. Due to the production of products with common machines, it is necessary to change the fixture and make adjustments, which take up a significant amount of time in the work shift and cause a loss of useful production time. On the other hand, the manufacturing company has different customers who tend to prioritize them and deliver the order based on their priority. One of the reasons for this prioritization is the monopoly of production and the small number of competitors.

Therefore, customers will be prioritized based on criteria such as deprivation, urgency of requests, purchase volume, and good credit. The product request is received by the customers without a prior plan and during the planning time and must be included in the production plan. The purpose of the research is to schedule production by minimizing the delivery time and reducing the total replacement times by considering the objective functions. A flexible job shop production schedule can be described below: A set of n tasks including j = {j1, j2,…, jn } can be processed on m machine as M = {M1, M2,…, Mm… }. Each task can be processed on multiple machines [8].

A flexible job shop production schedule determines the most suitable machine for each operation (Called machine selection) and the sequence of operations on the devices (Called the sequence of operations). The purpose of scheduling a flexible job shop schedule is to minimize some of the indicators, such as the completion time of the last work, the maximum delay, and the total flow time.

In addition, there are assumptions and limitations to flexible job shop production scheduling. Flexible job shop production schedules with a variety of optimization goals have been extensively studied in the literature. Some optimization goals are summarized in *Table 1*.

**Table 1. Objectives for optimizing flexible workshop production scheduling.**

| Symbol | Description of the Objective Function | Reference |
|---|---|---|
| $\max j(C_j)$ | The time interval from the beginning to the end of the work or the maximum time of completion of the work | [11], [12] |
| $\max(T_j)$ | Maximum tardiness | [16] |
| $\sum T_j$ | Total tardiness | [14], [15] |
| $(\Sigma T_j \left(\sum T_j\right)/n$ | Average of tardiness | [16], [17] |
| $\max(L_j)$ | Over maximum tardiness | [17] |
| $\sum I_j$ | Total unemployment time | [18] |
| $\sum F_j$ | Total process time (Flow) | [18] |
| $\left(\sum F_j\right)/n$ | Average workflow time | [16], [18] |
| $\max(W_j)$ | Maximum workload | [11] |
| $\sum W_j$ | Total work | [19], [20] |
| $\sum o_j$ | Total operating cost | [20], [21] |
| $\sum E_j$ | Total energy consumption | [21], [22] |

## 3.1 | Prioritization of Customers by Analytic Hierarchical Analysis Method

Several essential criteria were set for prioritizing customers by reading existing books, articles, publications, dissertations, and documents and consultation with six sales and business managers with more than ten years of experience. Five of the most important ones, which are described in detail below, were identified for the development of the pairwise comparison matrix questionnaire.

A questionnaire was provided to the experts, including 12 sales and business managers and six experienced people, for pairwise comparison. The collected information was entered into Excel software after collecting questionnaires and expert opinions. In the next step, to weigh each criterion, it was analyzed using the Analytic Hierarchical Analysis (AHP) model.

Five factors, according to experts, were identified as influential factors in customer prioritization among the various factors. They include timely settlement of invoices, purchase volume, participation in regional exhibitions, urgency of action, and deprivation of provinces.

In this criterion, the Atlas of Deprived Areas was used to identify deprived regions and cities better. It is worth mentioning that the ratio of the number of deprived areas to the total areas of the country was 56.8%. That is, more than half of the country's regions are less developed or deprived. The ranking is calculated based on different indicators in expert choice software using the hierarchical analysis method, and the results are as follows.

**Table 2. Prioritization of representatives based on research indicators.**

| Prioritization of Representatives Based on "Deprivation of Provinces" | | Prioritize Representatives Based on "Urgency (Action)" | | Prioritization of Representatives Based on "Participation in Regional Exhibitions" | | Prioritization of Representatives Based on "Purchase Volume" | | Prioritization of Representatives Based on "Timely Settlement of Bills" | |
|---|---|---|---|---|---|---|---|---|---|
| 0.022533 | Rep K | 0.027961 | Rep B | 0.023945 | Rep C | 0.018881 | Rep C | 0.017219 | Rep K |
| 0.022533 | Rep H | 0.027961 | Rep C | 0.024210 | Rep D | 0.020672 | Rep D | 0.025029 | Rep C |
| 0.022899 | Rep N | 0.027961 | Rep E | 0.026072 | Rep K | 0.026673 | Rep E | 0.025029 | Rep D |
| 0.022899 | Rep O | 0.027961 | Rep F | 0.040769 | Rep A | 0.026674 | Rep F | 0.040997 | Rep A |
| 0.037094 | Rep L | 0.027961 | Rep G | 0.040769 | Rep H | 0.043562 | Rep K | 0.040997 | Rep B |
| 0.037094 | Rep G | 0.027961 | Rep H | 0.041227 | Rep M | 0.045466 | Rep B | 0.040997 | Rep E |
| 0.057810 | Rep A | 0.027961 | Rep I | 0.041230 | Rep O | 0.048011 | Rep J | 0.040997 | Rep F |
| 0.061868 | Rep I | 0.027961 | Rep J | 0.041761 | Rep N | 0.064195 | Rep M | 0.040997 | Rep J |
| 0.061868 | Rep F | 0.027961 | Rep K | 0.067631 | Rep E | 0.066173 | Rep A | 0.070167 | Rep G |
| 0.061869 | Rep C | 0.029036 | Rep L | 0.068693 | Rep B | 0.070195 | Rep G | 0.070167 | Rep H |
| 0.068716 | Rep M | 0.049419 | Rep A | 0.068693 | Rep F | 0.070195 | Rep H | 0.070167 | Rep I |
| 0.105577 | Rep J | 0.060519 | Rep D | 0.113579 | Rep G | 0.108496 | Rep N | 0.114386 | Rep M |
| 0.105577 | Rep E | 0.203125 | Rep M | 0.113579 | Rep I | 0.108496 | Rep O | 0.114386 | Rep N |
| 0.155780 | Rep D | 0.203125 | Rep N | 0.115182 | Rep J | 0.114497 | Rep I | 0.114386 | Rep O |
| 0.155781 | Rep B | 0.203125 | Rep O | 0.166820 | Rep L | 0.167816 | Rep L | 0.174063 | Rep L |

**Table 3. Weight on indicator.**

| | |
|---|---|
| **Timely Settlement of Bills** | 0.164178 |
| **Purchase Volume** | 0.035337 |
| **Participate in Regional Exhibitions** | 0.035337 |
| **Emergency (Action)** | 0.519495 |
| **Deprivation of the Province** | 0.245652 |

Urgency has the highest weight among the five indicators.

**Table 4. Prioritization in order.**

| | |
|---|---|
| 0.05038 | **Rep A** |
| 0.02535 | **Rep K** |
| 0.03535 | **Rep C** |
| 0.0355 | **Rep H** |
| 0.03982 | **Rep F** |
| 0.04165 | **Rep G** |
| 0.0493 | **Rep I** |
| 0.05052 | **Rep E** |
| 0.05296 | **Rep J** |
| 0.06356 | **Rep B** |
| 0.0646 | **Rep L** |
| 0.0754 | **Rep D** |
| 0.13522 | **Rep O** |
| 0.13524 | **Rep N** |
| 0.14491 | **Rep M** |

## 3.2 | Proposed Algorithm

The following are assumed to be problem assumptions to make a research model:

I. The customer receives the whole order at once, and it is not possible to separate it.

II. Each machine can only process one task at a time.

III. New orders are received along the planning horizon besides the orders at the beginning of the planning horizon.

IV. There is no surplus production of orders.

V. There is no lost order.

VI. All devices are available in zero time.

VII. Each work is processed on one machine only during its processing time in each workstation, and there is no possibility of failure.

VIII. Not all machines are available continuously, and it takes time to set up to switch from one application to another to produce different products.

IX. Operations cannot be interrupted during the processing process.

## 3.3 | Particle Swarm Optimization Algorithm

The particle swarm optimization algorithm is described in this section among the stated metaheuristic algorithms. The particle swarm optimization algorithm is a population-based algorithm first introduced by Kennedy and Eberhart [23]. This algorithm quickly excelled among other algorithms.

The main reasons for this superiority are the simplicity of the main algorithm compared to different algorithms, the availability of its code, and the proof of its efficiency according to studies conducted with its low computational cost. The components of this algorithm are described in *Table 5*. Changing the position of particles within the search space concerning the socio-psychological orientation of individuals is done to imitate other people. The position of each particle changes depending on the experience of the neighbors and the particle itself.

If $\vec{x}_i(t)$ , indicates the position of the Pi particle at the moment or step t. Its position is obtained from the *Eq. (1)*. In this equation $\vec{v_i}(t)$ $\vec{v}_i(t)$ is called the velocity vector of the particle position change. $\vec{x}_i(t-1)\vec{x_i}(t-1)$ Indicates the current position of the particle.

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (1)$$

**Table 5. Components of particle swarm optimization algorithm.**

| Definition of Component | Algorithm Component |
|---|---|
| Algorithm community | Swarm |
| A feasible problem | Particle |
| The best personal memory of every particle in your memory | Pbest |
| The best memory of all the particles from the crowd | Gbest |
| Select a particle to guide particles to search space | Leader |
| Vector to determine the direction of particle motion to improve the answers | (V) |
| Control the previous motion memory feedback over the current motion of the particle | (W) |
| The degree of dependence of the new particle position on Pbest and Gbest | (C1, C2) |

$\vec{v}_i(t)$ is the amount of movement and its direction, $\vec{v_i}(t)$ is the reflection of particle social information among other particles. These are obtained from *Eq. (2)*. In this equation, r1, and r2 are random numbers between zero and one, C1 and C2 are the particle learning coefficients of the position of the best personal and optimal global memory, and W is the effect of the previous velocity vector on the new velocity vector.

$$\vec{v}_i(t) = W \times \vec{v}_i(t-1) + C_1 r_1 \left(\vec{x}_{Pbest}(t) - \vec{x}_i(t)\right) + C_2 r_2 \left(\vec{x}_{Gbest}(t) - \vec{x}_i(t)\right) \quad (2)$$

The algorithm used in this research is the Multiple Objective Particle Swarm Optimization (MOPSO) algorithm in discrete mode, which is the extended mode of the PSO algorithm in multi-objective mode.

## 3.4 | Multi-Objective Particle Swarm Algorithm

The PSO algorithm is a population-based algorithm. Hence, it is expected to obtain some unsatisfactory and varied answers each time the algorithm is executed. Algorithm development is presented using the research done by Reyes-Sierra and Coello [24]. It is important to note a few points in extending and generalizing the PSO algorithm to multi-objective mode:

I. How to choose particles as a leader[1] to prefer nondominated to dominated answers?

II. How to maintain nondominated answers over different generations?

III. How do we keep particle diversity[2] to avoid convergence into one answer?

Each particle can have different guides in multi-objective problems, one of which will be chosen to determine the new position. This set of directions is stored in a Repository (REP). The particles of this reservoir are used as a leader when changing position, which includes nondominated responses. Its members are updated at the end of each run of the algorithm.

The initial population is formed at the beginning of the algorithm, and nondominated members are then removed and stored in the REP. In the next step, REP members will be classified according to their quality. This classification is based on points on the uniformity order and diversity of REP members. A particle is chosen as the leader to perform the movement of a particle in each generation. The particle position is then evaluated, and the corresponding Pbest is updated.

If Pbest is defeated by a new particle or when the two are not comparable, in the sense that they do not defeat each other, it is replaced by a new answer. Then, all the particles of the REP population are updated. This process continues until a certain number of repetitions are completed. The multi-objective algorithm has two main differences compared to PSO:

**Selecting and updating guides**

- *How to choose guides from a set of non-dominated particles of equal importance. Can a guide be selected at random, or should other criteria be considered for decision-making?*
- *How to select particles to stay in the REP from one iteration to the next?*

**Creating a new answer**

- *How to promote the diversity of the two new response mechanisms, including position updating?*

There are various answers to the problem in multi-objective problems, all of which are good. Hence, the concept of guide in this algorithm is different from the classical PSO algorithm. All nondominated answers are selected as the candidate guide, and then one of them is chosen as the guide in many direct ways.

It is essential to know how to measure or classify REP members in this method. The density of responses in the target space is a possible way to measure the quality of REP members. It means how close the answers are to each other. Using an external memory is one of the direct ways nondominated solutions can be preserved over different generations.

An answer enters the REP when 1- it is not dominated by the REP members and 2- it dominates at least one of the REP members (The defeated members are removed from the REP). Given that the REP algorithm is updated each time it is run, the number of REP members increases. If the non-addition of non-dominated members is not controlled, it will increase the computational cost of the algorithm.

$O(kN^2)$ is the computational complexity of REP. N represents the number of REP members, and k represents the number of target functions that reach O (kMN2) if M is repeated. Therefore, the number of members of this memory should be limited.

Additional criteria need to be considered to limit space to decide on the maintenance of nondominated members when the tank is full. Researchers have developed various techniques for controlling memory in

[1] Leader

[2] Diversity

multi-objective evolutionary algorithms (Such as clustering and geographic designs that place nondominated members in cells).

Low-density homes are preferred over high-density homes when removing additional members from the REP. So, it is more likely to remove the answers that are in high-density houses. In this algorithm, the primary population is created first, then the guide of each member of the population is selected. Finally, the search process continues in repetitive steps.

The steps of the algorithm are as follows:

I. Creating primary populations and evaluating them.

II. Identifying nondominated members and saving them in REP.

III. Target space tabulation and determining the location of each REP member

IV. Each particle chose a guide from among the REP members and made its move.

V. The best personal memory of each particle is updated.

VI. The nondominated members of the current population will be added to the REP.

VII. We remove the nondominated members in REP.

VIII. Remove additional members if the number of REP members exceeds the specified capacity.

IX. Return to step 3 if the termination conditions are not met; otherwise, the termination is.

# 4 | Research Results

Products are not made for storage, and their construction starts only after receiving the order, as stated in the assumptions of the problem. Given that orders are entered instantaneously along the scheduling horizon, ignoring this issue will cause deviation from the program. The urgency of orthopedic procedures is the reason for accepting instant orders that are not predictable due to increased accidents. Manufacturing companies are also required to sell copies of their products if the operation is an emergency, with the establishment of the Health Transformation Plan by the Ministry of Health. It makes the company's production program more flexible. The algorithm is rerun after receiving each order, and the combination of previous incomplete orders and new orders is considered for re-planning.

The new order is also processed without adjusting the device program when similar products are being processed on one of the devices. It reduces the delay in product delivery. One of the main reasons for the delay is the limited number of devices in some workstations. According to the policy of increasing the company's production capacity, it is hoped that in 2021, the production capacity will double to reduce the number of delays significantly.

A numerical example is presented to express the performance of the proposed model in this section. A multi-objective particle swarm algorithm is used to solve this partially flexible job shop production problem. The parameters of the algorithm are expressed in the *Table 6*.

**Table 6. Parameters of the proposed algorithm.**

| Parameter | Quantity |
|---|---|
| Number of times the algorithm repeats | 200 |
| Population size | 100 |
| Number of tank members | 10 |
| C1 | 2.05 |
| C2 | 2.05 |

In this example, 20 pieces are processed, named from P1 to P20, by six machines from M1 to M6. The time available for producing parts is considered to be two work shifts and equivalent to 960 minutes. The configuration time of each machine is stated in *Table 7*. The very long configuration time shows the importance of this model.

Also, the average amount of time required to process parts is shown in *Table 8*. Parts P1 to P4 can be processed with all machines except M1 and have different processing times on each of them. Also, parts P5 to P20 can only be processed with the M1 machine.

**Table 7. Setup time on different machines (Minutes).**

| Raw | Type of Machine | Setup Time | Available Time |
|---|---|---|---|
| 1 | M1 | 90 | 960 |
| 2 | M2 | 420 | 960 |
| 3 | M3 | 420 | 960 |
| 4 | M4 | 120 | 960 |
| 5 | M5 | 120 | 960 |
| 6 | M6 | 120 | 960 |

**Table 8. Time required to process parts in different machines (Minutes).**

| Product | M1 | M2 | M3 | M4 | M5 | M6 |
|---|---|---|---|---|---|---|
| P1 | | 15 | 15 | 20 | 20 | 20 |
| P2 | | 10 | 10 | 14 | 14 | 14 |
| P3 | | 10 | 10 | 14 | 14 | 14 |
| P4 | | 10 | 10 | 14 | 14 | 14 |
| P5 | 30 | 0 | 0 | 0 | 0 | 0 |
| P6 | 30 | 0 | 0 | 0 | 0 | 0 |
| P7 | 30 | 0 | 0 | 0 | 0 | 0 |
| P8 | 20 | 0 | 0 | 0 | 0 | 0 |
| P9 | 20 | 0 | 0 | 0 | 0 | 0 |
| P10 | 30 | 0 | 0 | 0 | 0 | 0 |
| P11 | 30 | 0 | 0 | 0 | 0 | 0 |
| P12 | 35 | 0 | 0 | 0 | 0 | 0 |
| P13 | 30 | 0 | 0 | 0 | 0 | 0 |
| P14 | 30 | 0 | 0 | 0 | 0 | 0 |
| P15 | 30 | 0 | 0 | 0 | 0 | 0 |
| P16 | 25 | 0 | 0 | 0 | 0 | 0 |
| P17 | 25 | 0 | 0 | 0 | 0 | 0 |
| P18 | 25 | 0 | 0 | 0 | 0 | 0 |
| P19 | 25 | 0 | 0 | 0 | 0 | 0 |
| P20 | 25 | 0 | 0 | 0 | 0 | 0 |

In total, 167 orders were received from 44 customers. Customers' orders were received within one month of the planning horizon. They are reprogrammed every day if there is a new order.

Orders received from 44 customers within one month are listed in the appendices of the article in *Table 1*. Column (1) in this table states the order number, which is from 1 to 167. Column (2) displays the customer number, and column (3) lists the ordered part from P1 to P20. Column (4) states the number of requested features. Column (5) indicates the order day so that no order is received in the first three days and the first order is received on the fourth day.

All customers have an immediate request for their order, so the delivery time for all orders is zero. Also, the demands of the previous month are not considered in this numerical example. If there is an order left from the last month, it can be regarded as a first-day order. The customer weight is calculated based on the criteria in column (6) and distributed in the range [1,4]. Column (7) is the result of the answer of the proposed algorithm that shows the delay in delivery. This numerical example is performed 50 times, and the values for the total delay time (Z1) and the total setting time (Z2) are shown in *Table 9*.

**Table 9. Results of 50 numerical examples.**

| | Average | Standard Deviation | Max | Min |
|---|---|---|---|---|
| Z1 | 268.7 | 20.5 | 291.1 | 230.1 |
| Z2 | 9639 | 518.4 | 10350 | 8580 |

The program obtained for each of the machines is shown in *Table 1* of the Appendices in column 9. This result is related to the situation that the setting time is 9180 minutes, and the total weight delay is equal to 275.32 days. The values obtained were also compared with the manufacturer's actual program, and the results show a 13% improvement. Each customer's order delay time is descending the weight coefficient of customers, as shown in *Fig. 1*. Attempts have been made to deliver essential customer orders with less delay, as shown in the figure. The figure shows the downward trend in reducing delays with increasing customer importance. The stated problem to evaluate the model's performance has been solved 50 times considering only one of the objective functions.

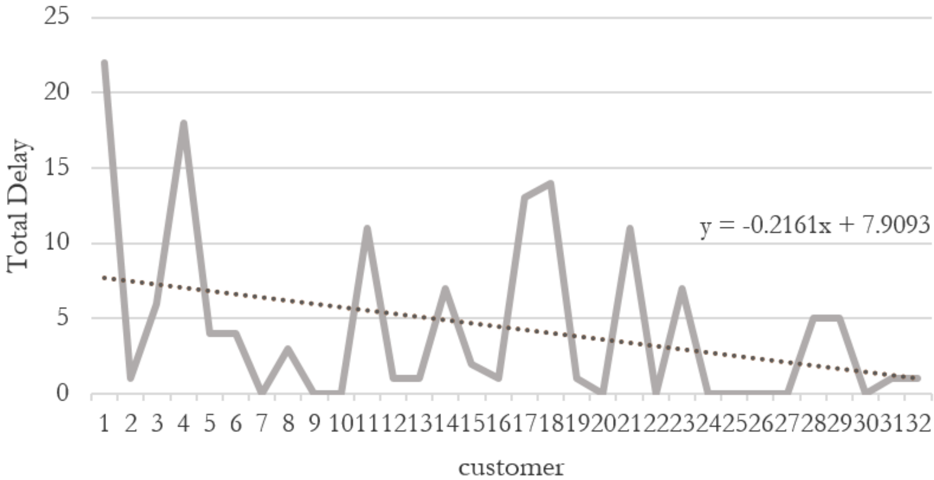


**Fig. 1. The amount of delay in customer order in descending order of customer weight.**

The best value for the total weight of the delays is 205.8, considering the objective function Z1. The entire machine setting time is 15270 minutes in this mode. The best value for orders received is 7650 minutes; if there is no delay in delivering orders, only a reduction in the total replacement times (Z2) is considered. The total weight of delays is 1/808.

Therefore, it is observed that a lot of time is lost for multiple settings if only minimizing delivery delays is considered. Delivery of orders will be delayed if only the reduction of settings is taken into account. The results obtained from solving the problem show that in a situation where the adjustment time is significant and the available time of the machines is significantly reduced, the proposed model can provide a more realistic production program by considering the stated objective functions and prevent many changes in the program.

# 5 | Summary

A review of the process used by the production planning manager is the first step in solving flexible job-shop scheduling problems. Workshop entries are a set of constantly received orders on the horizon and must be delivered in order of priority. Determining the order of work done by that machine is the second step in solving flexible job-shop scheduling problems. The third step is to compare the available methods and choose the best solution method according to the situation.

In this research, a partially flexible job shop production planning problem was investigated. One of the apparent features of this issue is the simultaneous examination of objective functions to minimize delay in delivery and minimize device configuration time, which is ignored in most studies. Instant orders and rescheduling were also reviewed. It developed a multi-objective particle cluster optimization algorithm to solve a problem. The proposed model with various examples has been evaluated. The results show the efficiency of the proposed algorithm. In the present study, it is assumed that the whole order should be processed together. Besides, it seems that the results of the problem will be better in future studies if partial processing of works is possible. High-consumption parts storage can significantly reduce lost time. The variability of machining time of operations related to a type of part due to the gradual wear of tools or other factors can be considered in the formulation of the problem.

# Conflict of Interest

The authors declare no conflict of interest.

# Data Availability

All data are included in the text.

# Funding

This research received no specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

# Reference

[1] Petrović, M., Vuković, N., Mitić, M., & Miljković, Z. (2016). Integration of process planning and scheduling using chaotic particle swarm optimization algorithm. *Expert systems with applications*, *64*, 569–588. https://doi.org/10.1016/j.eswa.2016.08.019

[2] Chen, R., Yang, B., Li, S., & Wang, S. (2020). A self-learning genetic algorithm based on reinforcement learning for flexible job-shop scheduling problem. *Computers & industrial engineering*, *149*, 106778. https://doi.org/10.1016/j.cie.2020.106778

[3] Akyol, D. E., & Bayhan, G. M. (2007). A review on evolution of production scheduling with neural networks. *Computers & industrial engineering*, *53*(1), 95–122. https://doi.org/10.1016/j.cie.2007.04.006

[4] Gonçalves, J. F., De Magalhães Mendes, J. J., & Resende, M. G. C. (2005). A hybrid genetic algorithm for the job shop scheduling problem. *European journal of operational research*, *167*(1), 77–95. https://doi.org/10.1016/j.ejor.2004.03.012

[5] Çaliş, B., & Bulkan, S. (2015). A research survey: Review of AI solution strategies of job shop scheduling problem. *Journal of intelligent manufacturing*, *26*(5), 961–973. https://doi.org/10.1007/s10845-013-0837-8

[6] Li, X., Peng, Z., Du, B., Guo, J., Xu, W., & Zhuang, K. (2017). Hybrid artificial bee colony algorithm with a rescheduling strategy for solving flexible job shop scheduling problems. *Computers & industrial engineering*, *113*, 10–26. https://doi.org/10.1016/j.cie.2017.09.005

[7] Amjad, M. K., Butt, S. I., Kousar, R., Ahmad, R., Agha, M. H., Faping, Z., Anjum, N., & Asgher, U. (2018). Recent research trends in genetic algorithm based flexible job shop scheduling problems. *Mathematical problems in engineering*, *2018*. https://doi.org/10.1155/2018/9270802

[8] Gao, K., Cao, Z., Zhang, L., Chen, Z., Han, Y., & Pan, Q. (2019). A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. *IEEE/CAA journal of automatica sinica*, *6*(4), 904–916. https://doi.org/10.1109/JAS.2019.1911540

[9] Özgüven, C., Özbakır, L., & Yavuz, Y. (2010). Mathematical models for job-shop scheduling problems with routing and process plan flexibility. *Applied mathematical modelling*, *34*(6), 1539–1548. https://doi.org/10.1016/j.apm.2009.09.002

[10] Özgüven, C., Yavuz, Y., & Özbakır, L. (2012). Mixed integer goal programming models for the flexible job-shop scheduling problems with separable and non-separable sequence dependent setup times. *Applied mathematical modelling*, *36*(2), 846–858. https://doi.org/10.1016/j.apm.2011.07.037

[11] Jahromi, M., & Tavakkoli-Moghaddam, R. (2012). A novel 0-1 linear integer programming model for dynamic machine-tool selection and operation allocation in a flexible manufacturing system. *Journal of manufacturing systems*, *31*(2), 224–231. https://doi.org/10.1016/j.jmsy.2011.07.008

[12] Shen, X. N., Han, Y., & Fu, J. Z. (2017). Robustness measures and robust scheduling for multi-objective stochastic flexible job shop scheduling problems. *Soft computing*, *21*, 6531–6554. https://doi.org/10.1007/s00500-016-2245-4

[13] Shen, X. N., & Yao, X. (2015). Mathematical modeling and multi-objective evolutionary algorithms applied to dynamic flexible job shop scheduling problems. *Information sciences*, *298*, 198–224. https://doi.org/10.1016/j.ins.2014.11.036

[14] Homayouni, S. M., Fontes, D. B. M. M., & Gonçalves, J. F. (2020). A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation. *International transactions in operational research*. https://doi.org/10.1111/itor.12878

[15] Özkul, A. O., Küçük, G., Çelik, C., Öztuna, N., Demirkan, M., Nizam, E. Ç., Yüksel, N., & Toy, A. Ö. (2021). An implementation of flexible job shop scheduling problem in a metal processing company. *Digital conversion on the way to industry 4.0: Selected papers from ISPR2020, online-Turkey* (pp. 817–830). Springer. https://doi.org/10.1007/978-3-030-62784-3_68

[16] Zhang, G., Gao, L., & Shi, Y. (2011). An effective genetic algorithm for the flexible job-shop scheduling problem. *Expert systems with applications*, *38*(4), 3563–3573. https://doi.org/10.1016/j.eswa.2010.08.145

[17] Pezzella, F., Morganti, G., & Ciaschetti, G. (2008). A genetic algorithm for the flexible job-shop scheduling problem. *Computers & operations research*, *35*(10), 3202–3212. https://doi.org/10.1016/j.cor.2007.02.014

[18] Baykasoglu, A. (2002). Linguistic-based meta-heuristic optimization model for flexible job shop scheduling. *International journal of production research*, *40*(17), 4523–4543. https://doi.org/10.1080/00207540210147043

[19] Scrich, C. R., Armentano, V. A., & Laguna, M. (2004). Tardiness minimization in a flexible job shop: A tabu search approach. *Journal of intelligent manufacturing*, *15*(1), 103–115. https://doi.org/10.1023/B:JIMS.0000010078.30713.e9

[20] Brandimarte, P. (1993). Routing and scheduling in a flexible job shop by tabu search. *Annals of operations research*, *41*(3), 157–183. https://doi.org/10.1007/BF02023073

[21] Tay, J. C., & Ho, N. B. (2008). Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems. *Computers & industrial engineering*, *54*(3), 453–473. https://doi.org/10.1016/j.cie.2007.08.008

[22] Chen, J. C., Wu, C. C., Chen, C. W., & Chen, K. H. (2012). Flexible job shop scheduling with parallel machines using genetic algorithm and grouping genetic algorithm. *Expert systems with applications*, *39*(11), 10016–10021. https://doi.org/10.1016/j.eswa.2012.01.211

[23] Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. *1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation* (Vol. 5, pp. 4104–4108). IEEE. https://doi.org/10.1109/ICSMC.1997.637339

[24] Reyes-Sierra, M., Coello, C. A. C. (2006). Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International journal of computational intelligence research*, *2*(3), 287–308. https://B2n.ir/ss2637

# Appendix

**Table A1. Customer orders within one month.**

| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6 | Column 7 | Column 8 | Column 9 |
|---|---|---|---|---|---|---|---|---|
| Raw | Customer No | Type of Product | Quantity | Order Day | Customer Weight | Delay in Delivery (Day) | Actual Delay Rate (Day) | Machine |
| 1 | 1 | P6 | 8 | 4 | 1.000 | 2 | 3 | M1 |
| 2 | 1 | P17 | 3 | 4 | 1.000 | 0 | 1 | M1 |
| 3 | 27 | P7 | 3 | 4 | 2.322 | 0 | 1 | M1 |
| 4 | 27 | P13 | 9 | 4 | 2.322 | 0 | 0 | M1 |
| 5 | 27 | P17 | 6 | 4 | 2.322 | 0 | 0 | M1 |
| 6 | 27 | P15 | 2 | 4 | 2.322 | 0 | 1 | M1 |
| 7 | 11 | P8 | 3 | 4 | 1.423 | 1 | 1 | M1 |
| 8 | 1 | P3 | 192 | 4 | 1.000 | 2 | 3 | M5 |
| 9 | 1 | P1 | 192 | 4 | 1.000 | 3 | 3 | M6 |
| 10 | 44 | P6 | 6 | 5 | 4.000 | 0 | 1 | M1 |
| 11 | 44 | P17 | 3 | 5 | 4.000 | 0 | 2 | M1 |
| 12 | 38 | P9 | 8 | 5 | 3.827 | 0 | 0 | M1 |
| 13 | 18 | P2 | 40 | 5 | 2.279 | 0 | 0 | M2 |
| 14 | 18 | P3 | 25 | 5 | 2.279 | 0 | 0 | M4 |

**Table A1. Continued.**

| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6 | Column 7 | Column 8 | Column 9 |
|---|---|---|---|---|---|---|---|---|
| Raw | Customer No | Type of Product | Quantity | Order Day | Customer Weight | Delay in Delivery (Day) | Actual Delay Rate (Day) | Machine |
| 15 | 18 | P1 | 65 | 5 | 2.279 | 1 | 1 | M4 |
| 16 | 6 | P20 | 2 | 7 | 1.387 | 0 | 0 | M1 |
| 17 | 24 | P2 | 20 | 7 | 2.322 | 0 | 0 | M2 |
| 18 | 24 | P1 | 20 | 7 | 2.322 | 0 | 0 | M4 |
| 19 | 4 | P6 | 4 | 7 | 1.000 | 0 | 1 | M1 |
| 20 | 8 | P15 | 1 | 7 | 1.423 | 0 | 1 | M1 |
| 21 | 27 | P2 | 15 | 7 | 2.322 | 0 | 0 | M2 |
| 22 | 27 | P3 | 8 | 7 | 2.322 | 0 | 0 | M5 |
| 23 | 27 | P1 | 23 | 7 | 2.322 | 0 | 0 | M4 |
| 24 | 2 | P3 | 50 | 7 | 1.000 | 0 | 0 | M5 |
| 25 | 2 | P1 | 50 | 7 | 1.000 | 1 | 1 | M4 |
| 26 | 18 | P11 | 3 | 8 | 2.279 | 0 | 0 | M1 |
| 27 | 18 | P9 | 11 | 8 | 2.279 | 0 | 0 | M1 |
| 28 | 29 | P3 | 10 | 9 | 2.329 | 0 | 0 | M5 |
| 29 | 29 | P1 | 10 | 9 | 2.329 | 0 | 0 | M4 |
| 30 | 4 | P6 | 2 | 9 | 1.000 | 0 | 1 | M1 |
| 31 | 20 | P13 | 2 | 9 | 2.298 | 0 | 1 | M1 |
| 32 | 12 | P8 | 2 | 9 | 1.423 | 0 | 0 | M1 |
| 33 | 12 | P6 | 3 | 9 | 1.423 | 0 | 0 | M1 |
| 34 | 12 | P17 | 1 | 9 | 1.423 | 0 | 0 | M1 |
| 35 | 35 | P4 | 25 | 9 | 3.827 | 0 | 0 | M4 |
| 36 | 35 | P1 | 25 | 9 | 3.827 | 0 | 0 | M6 |
| 37 | 35 | P16 | 8 | 9 | 3.827 | 0 | 1 | M1 |
| 38 | 14 | P9 | 3 | 11 | 1.423 | 0 | 0 | M1 |
| 39 | 14 | P16 | 3 | 11 | 1.423 | 0 | 0 | M1 |
| 40 | 3 | P3 | 212 | 11 | 1.000 | 3 | 4 | M5 |
| 41 | 3 | P1 | 212 | 11 | 1.000 | 3 | 3 | M3 |
| 42 | 34 | P19 | 1 | 12 | 3.827 | 0 | 1 | M1 |
| 43 | 34 | P20 | 1 | 12 | 3.827 | 0 | 0 | M1 |
| 44 | 27 | P3 | 20 | 12 | 2.322 | 2 | 2 | M5 |
| 45 | 27 | P1 | 20 | 12 | 2.322 | 0 | 0 | M6 |
| 46 | 32 | P17 | 2 | 12 | 3.459 | 0 | 0 | M1 |
| 47 | 32 | P12 | 1 | 12 | 3.459 | 0 | 0 | M1 |
| 48 | 8 | P16 | 2 | 12 | 1.423 | 0 | 1 | M1 |
| 49 | 6 | P8 | 3 | 12 | 1.387 | 1 | 1 | M1 |
| 50 | 6 | P13 | 5 | 12 | 1.387 | 1 | 1 | M1 |
| 51 | 6 | P16 | 2 | 12 | 1.387 | 0 | 1 | M1 |
| 52 | 27 | P11 | 2 | 12 | 2.322 | 0 | 0 | M1 |
| 53 | 27 | P13 | 2 | 12 | 2.322 | 0 | 0 | M1 |
| 54 | 11 | P20 | 1 | 13 | 1.423 | 0 | 1 | M1 |
| 55 | 11 | P19 | 2 | 13 | 1.423 | 0 | 0 | M1 |
| 56 | 16 | P16 | 1 | 13 | 2.268 | 1 | 1 | M1 |
| 57 | 16 | P20 | 2 | 13 | 2.268 | 0 | 0 | M1 |
| 58 | 16 | P19 | 1 | 13 | 2.268 | 0 | 0 | M1 |
| 59 | 37 | P11 | 10 | 13 | 3.827 | 0 | 0 | M1 |
| 60 | 43 | P3 | 77 | 13 | 3.889 | 0 | 0 | M2 |
| 61 | 43 | P1 | 77 | 13 | 3.889 | 1 | 1 | M6 |
| 62 | 20 | P3 | 125 | 13 | 2.298 | 2 | 2 | M2 |
| 63 | 20 | P1 | 125 | 13 | 2.298 | 3 | 3 | M3 |
| 64 | 4 | P13 | 6 | 13 | 1.000 | 1 | 1 | M1 |
| 65 | 4 | P17 | 4 | 13 | 1.000 | 1 | 1 | M1 |
| 66 | 4 | P6 | 4 | 13 | 1.000 | 1 | 1 | M1 |
| 67 | 6 | P13 | 2 | 18 | 1.387 | 0 | 0 | M1 |
| 68 | 8 | P8 | 3 | 18 | 1.423 | 1 | 1 | M1 |
| 69 | 8 | P17 | 6 | 18 | 1.423 | 1 | 1 | M1 |
| 70 | 8 | P18 | 1 | 18 | 1.423 | 1 | 2 | M1 |
| 71 | 31 | P7 | 8 | 18 | 3.114 | 0 | 0 | M1 |
| 72 | 31 | P5 | 10 | 18 | 3.114 | 0 | 0 | M1 |
| 73 | 31 | P11 | 2 | 18 | 3.114 | 0 | 0 | M1 |
| 74 | 23 | P3 | 124 | 19 | 2.300 | 1 | 1 | M2 |
| 75 | 23 | P3 | 124 | 19 | 2.300 | 1 | 1 | M6 |
| 76 | 4 | P15 | 6 | 19 | 1.000 | 2 | 2 | M1 |
| 77 | 4 | P6 | 2 | 19 | 1.000 | 0 | 1 | M1 |
| 78 | 11 | 12 | 2 | 19 | 1.423 | 0 | 0 | M1 |
| 79 | 38 | 10 | 17 | 19 | 3.827 | 5 | 5 | M1 |

**Table A1. Continued.**

| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6 | Column 7 | Column 8 | Column 9 |
|---|---|---|---|---|---|---|---|---|
| Raw | Customer No | Type of Product | Quantity | Order Day | Customer Weight | Delay in Delivery (Day) | Actual Delay Rate (Day) | Machine |
| 80 | 8 | 6 | 10 | 19 | 1.423 | 1 | 1 | M1 |
| 81 | 44 | 17 | 2 | 19 | 4.000 | 0 | 0 | M1 |
| 82 | 20 | P3 | 115 | 19 | 2.298 | 1 | 1 | M5 |
| 83 | 20 | P1 | 115 | 19 | 2.298 | 1 | 1 | M3 |
| 84 | 30 | P13 | 61 | 19 | 3.109 | 11 | 12 | M1 |
| 85 | 41 | P11 | 2 | 20 | 3.860 | 0 | 1 | M1 |
| 86 | 32 | P17 | 1 | 20 | 3.459 | 0 | 0 | M1 |
| 86 | 1 | P6 | 9 | 20 | 1.000 | 8 | 8 | M1 |
| 87 | 1 | P13 | 8 | 20 | 1.000 | 0 | 0 | M1 |
| 88 | 8 | P17 | 2 | 20 | 1.423 | 0 | 0 | M1 |
| 89 | 13 | P12 | 1 | 20 | 1.423 | 0 | 0 | M1 |
| 90 | 27 | P20 | 2 | 21 | 2.322 | 0 | 0 | M1 |
| 91 | 27 | P8 | 1 | 21 | 2.322 | 2 | 2 | M1 |
| 92 | 27 | P13 | 6 | 21 | 2.322 | 0 | 0 | M1 |
| 93 | 27 | P11 | 2 | 21 | 2.322 | 3 | 4 | M1 |
| 94 | 4 | P10 | 1 | 21 | 1.000 | 0 | 0 | M1 |
| 95 | 4 | P15 | 1 | 21 | 1.000 | 1 | 1 | M1 |
| 96 | 4 | P6 | 1 | 21 | 1.000 | 0 | 0 | M6 |
| 97 | 24 | P2 | 20 | 21 | 2.322 | 0 | 0 | M2 |
| 98 | 24 | P3 | 34 | 21 | 2.322 | 0 | 0 | M3 |
| 99 | 24 | P1 | 54 | 21 | 2.322 | 0 | 0 | M1 |
| 100 | 24 | P11 | 3 | 21 | 2.322 | 1 | 1 | M1 |
| 101 | 24 | P6 | 5 | 21 | 2.322 | 3 | 3 | M1 |
| 102 | 1 | P6 | 11 | 21 | 1.000 | 0 | 0 | M1 |
| 103 | 41 | P11 | 2 | 21 | 3.860 | 0 | 1 | M1 |
| 104 | 20 | P17 | 1 | 21 | 2.298 | 0 | 0 | M5 |
| 105 | 33 | P3 | 16 | 22 | 3.827 | 0 | 0 | M3 |
| 106 | 33 | P1 | 16 | 22 | 3.827 | 0 | 0 | M1 |
| 107 | 20 | P11 | 1 | 22 | 2.298 | 5 | 6 | M1 |
| 108 | 40 | P13 | 18 | 22 | 3.827 | 0 | 0 | M1 |
| 109 | 11 | P18 | 2 | 22 | 1.423 | 0 | 0 | M1 |
| 110 | 11 | P6 | 2 | 22 | 1.423 | 1 | 1 | M5 |
| 111 | 8 | P18 | 1 | 22 | 1.423 | 0 | 2 | M1 |
| 112 | 31 | P6 | 2 | 22 | 3.114 | 0 | 0 | M1 |
| 113 | 27 | P20 | 1 | 22 | 2.322 | 0 | 0 | M1 |
| 114 | 27 | P8 | 1 | 22 | 2.322 | 0 | 0 | M1 |
| 115 | 27 | P13 | 6 | 22 | 2.322 | 1 | 1 | M1 |
| 116 | 27 | P11 | 2 | 22 | 2.322 | 0 | 0 | M1 |
| 117 | 27 | P20 | 1 | 22 | 2.322 | 0 | 2 | M1 |
| 118 | 27 | P3 | 12 | 23 | 2.322 | 0 | 0 | M5 |
| 119 | 27 | P1 | 12 | 23 | 2.322 | 0 | 1 | M3 |
| 120 | 43 | P3 | 13 | 23 | 3.889 | 0 | 0 | M5 |
| 121 | 43 | P1 | 13 | 23 | 3.889 | 0 | 0 | M3 |
| 122 | 32 | P6 | 1 | 23 | 3.459 | 0 | 0 | M1 |
| 123 | 6 | P6 | 1 | 23 | 1.387 | 0 | 0 | M1 |
| 124 | 14 | P2 | 20 | 23 | 1.423 | 0 | 0 | M6 |
| 125 | 14 | P18 | 9 | 23 | 1.423 | 3 | 4 | M1 |
| 126 | 14 | P16 | 6 | 23 | 1.423 | 3 | 3 | M1 |
| 127 | 14 | P13 | 14 | 23 | 1.423 | 0 | 0 | M1 |
| 128 | 14 | P11 | 2 | 23 | 1.423 | 3 | 3 | M1 |
| 129 | 14 | P17 | 6 | 23 | 1.423 | 2 | 2 | M1 |
| 130 | 4 | P6 | 5 | 25 | 1.000 | 1 | 1 | M1 |
| 131 | 4 | P15 | 2 | 25 | 1.000 | 1 | 1 | M1 |
| 132 | 38 | P6 | 10 | 25 | 3.827 | 0 | 0 | M1 |
| 133 | 35 | P3 | 30 | 25 | 3.827 | 0 | 0 | M5 |
| 134 | 35 | P1 | 30 | 25 | 3.827 | 0 | 0 | M3 |
| 135 | 35 | P17 | 8 | 25 | 3.827 | 0 | 2 | M1 |
| 136 | 27 | P6 | 1 | 26 | 2.322 | 0 | 0 | M1 |
| 137 | 27 | P13 | 5 | 26 | 2.322 | 2 | 2 | M1 |
| 138 | 4 | P6 | 6 | 27 | 1.000 | 0 | 0 | M1 |
| 139 | 8 | P6 | 2 | 27 | 1.423 | 0 | 1 | M1 |
| 140 | 28 | P2 | 83 | 27 | 2.322 | 0 | 0 | M2 |
| 141 | 28 | P1 | 83 | 27 | 2.322 | 1 | 1 | M3 |
| 142 | 4 | P15 | 2 | 28 | 1.000 | 3 | 4 | M1 |
| 143 | 27 | P3 | 50 | 28 | 2.322 | 0 | 0 | M5 |

**Table A1. Continued.**

| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6 | Column 7 | Column 8 | Column 9 |
|---|---|---|---|---|---|---|---|---|
| Raw | Customer No | Type of Product | Quantity | Order Day | Customer Weight | Delay in Delivery (Day) | Actual Delay Rate (Day) | Machine |
| 144 | 27 | P1 | 50 | 28 | 2.322 | 1 | 1 | M3 |
| 145 | 27 | P6 | 5 | 28 | 2.322 | 4 | 5 | M1 |
| 146 | 27 | P13 | 2 | 28 | 2.322 | 0 | 0 | M1 |
| 147 | 27 | P12 | 2 | 28 | 2.322 | 2 | 2 | M1 |
| 148 | 10 | P3 | 30 | 29 | 1.423 | 0 | 1 | M5 |
| 149 | 10 | P1 | 30 | 29 | 1.423 | 0 | 0 | M3 |
| 150 | 4 | P11 | 2 | 29 | 1.000 | 4 | 5 | M1 |
| 151 | 11 | P8 | 1 | 29 | 1.423 | 2 | 3 | M1 |
| 152 | 6 | P6 | 4 | 29 | 1.387 | 2 | 2 | M1 |
| 153 | 32 | P15 | 8 | 29 | 3.459 | 2 | 2 | M1 |
| 154 | 32 | P13 | 10 | 29 | 3.459 | 3 | 3 | M1 |
| 155 | 32 | P6 | 6 | 29 | 3.459 | 2 | 2 | M1 |
| 156 | 6 | P6 | 4 | 30 | 1.387 | 0 | 0 | M1 |
| 157 | 24 | P2 | 20 | 30 | 2.322 | 0 | 0 | M2 |
| 158 | 24 | P3 | 12 | 30 | 2.322 | 0 | 0 | M5 |
| 159 | 24 | P1 | 32 | 30 | 2.322 | 0 | 1 | M3 |
| 160 | 6 | P6 | 4 | 30 | 1.387 | 0 | 0 | M1 |
| 161 | 26 | P11 | 11 | 30 | 2.322 | 3 | 4 | M1 |
| 162 | 26 | P17 | 15 | 30 | 2.322 | 3 | 3 | M1 |
| 163 | 26 | P15 | 29 | 30 | 2.322 | 5 | 6 | M1 |
| 164 | 26 | P6 | 10 | 30 | 2.322 | 2 | 2 | M1 |
| 165 | 1 | P11 | 25 | 30 | 1.000 | 4 | 4 | M1 |
| 166 | 44 | P6 | 4 | 30 | 4.000 | 1 | 2 | M1 |
| 167 | 44 | P17 | 1 | 30 | 4.000 | 0 | 1 | M1 |