



## SIMULATION OF JOB SEQUENCING FOR STOCHASTIC SCHEDULING WITH A GENETIC ALGORITHM

Prasad Bari <sup>1,2\*</sup>, Prasad Karande <sup>1</sup>, Jayston Menezes <sup>2</sup>

<sup>1</sup> Department of Mechanical Engineering, Veermata Jijabai Technological Institute, Mumbai, India

<sup>2</sup> Department of Mechanical Engineering, Fr. C. Rodrigues Institute of Technology, Vashi, Navi-Mumbai, India

Received: 15 April 2022

Accepted: 18 June 2022

First online: 06 July 2022

Research paper

**Abstract:** Sequencing is done to determine the order in which the jobs are to be processed. Extensive research has been carried out with an aim to tackle real-world scheduling problems. In industries, experimentation is performed before an ultimate choice is made to know the optimal priority sequencing rule. Therefore, an extensive approach to selecting the correct choice is necessary for the management decision-making perspective. In this research, the genetic algorithm (GA) and working of a simulation environment are explained, in which a scheduling operator, under any given circumstances, can obtain the appropriate sequence for job scheduling in a shop. The paper also explains the stochastic based linguistic, scenarios and probabilistic approaches to solve sequencing problem. The simulation environment allows the operator to select the tardiness and non-tardiness related performance measures. The simulator takes input values such as number of jobs, processing time and due date and discovers a near-optimal sequence for scheduling of jobs that minimizes the performance measures selected by the operator as per requirement. The case study considered is solved using scenarios based stochastic scheduling approach and results are shown. The results are compared with the classical method used in the company and observed that the proposed approach gives a better result.

**Key words:** Stochastic scheduling, Genetic algorithm, Sequencing, Tardiness

### 1. Introduction

Job scheduling defines the order in which jobs are to be completed at one or more workplaces in a workshop. Job scheduling is critical to avoid long lines or production delays, which can result in financial losses or penalties for the firm. Scheduling of the

\* Corresponding author.

prasad.bari@fcrit.ac.in (P. Bari), pmkarande@me.vjti.ac.in (P. Karande),

jaystonmenezes@gmail.com (J. Menezes)

jobs can be done by keeping the main focus on one of the factors like priority sequencing rules (Sharma & Jain, 2015), performance measures (Oyetunji, 2009), cost minimization (Bari & Karande, 2020)(Bari & Karande, 2022), preference to weighted jobs, etc. Oyetunji (2009) formulated expressions of performance measures for computing their values. In this paper, the primary focus is on the minimization of the performance measures for scheduling. Performance measures can be minimized by arranging jobs in a particular sequence. In order to obtain the optimum sequence of the jobs, it is essential to know the number of jobs, their corresponding processing time (PT), and their due date (DD).

A job sequence is created by applying sequencing rules such as shortest processing time (SPT), in which the job that requires the least amount of machine processing time is scheduled first. It is applicable in the need of reduction of average completion time. Under the earliest due date (EDD) rule, jobs are completed in the order of their earlier due date. It can be used to cut down on tardiness. Jobs are processed in the order they arrive on the machine under the first-come, first-served (FCFS) rule. It is effective at lowering the variance in completion time for any dataset. The operator follows one of the sequencing rules as per their industry requirement. The operator's primary focus is primarily on the performance measure values of the sequence so that he can process the operation in the lowest possible time. In the scheduling operations, Kumar et al. (2017) discussed about 13 significant performance measures: completion time, flow time, total flow time, average job completion time, average number of jobs in the system, percentage utilization, lateness, average lateness, tardiness, total tardiness, maximum tardiness, average tardiness, and number of tardy jobs. Section 2 explains these performance measures in detail. To maximize production efficiency, all performance measurements except the percentage utilization from the considered performance measures should be minimized. Maximum percentage utilization indicates that the processing of the job is being done effectively on the machines in the job shop. Applying a PROMETHEE-GAIA technique can optimize these performance measures (Bari & Karande, 2021).

Considering the deterministic scheduling model, the optimum sequence with reference to these performance measures can be attained. However, in real-world scheduling problems, various factors affect the PT of the jobs that cannot be neglected. Therefore, such problems can be solved using the stochastic model. The stochastic approaches are more significant because of their capability to handle non-linear and multi-objective formulations effortlessly. The stochastic model can be solved using either the exact method or heuristic approach. However, the computational time required to solve deterministic and stochastic problems using the exact method is more than the heuristic approach. One of the popular heuristic methods is the genetic algorithm (GA). The trend for using stochastic techniques with GA is increasing for solving industrialized problems because GA finds a better objective function value for each iteration in less computational time. (Deb et al., 2002), (Sarkar & Modak, 2005), (Koratiya et al., 2010), (Ramteke & Srinivasan, 2012). Shrouf et al. (2014) proposed the application of GA to solve the single machine scheduling in the time off use tariff with respect to the machine's status, which consists of processing, idle, turning on and turning off. Roy et al. (2019) introduced novel GA concerning selection and crossover operation and found it effective to solve the travelling salesman problem. They have also mentioned that the algorithm proved effective for solving other problems like network optimization. Kurniawan et al. (2020) implemented GA to determine the job

sequencing, the job assignment, and the starting time of the job. Bayu et al. (2020) applied GA by considering stochastic conditions for sequencing the operations of gasoline blending and distribution plants to give gasoline a high commercial latent without negotiating quality and the clients' demand. Stanković et al. (2020) presented a model for solving flexible job shop scheduling problem (FJSP) built on meta-heuristic algorithms, tabu search, GA, and ant colony optimization. They have demonstrated the effectiveness of the GA method in resolving the FJSP problem, which gives commendatory results after testing. Garg (2016, 2019) solved the constraint optimization problem by feeding genetic operators of GA with particle swarm optimization and a gravitational search algorithm and discovered that these combined algorithms were efficient in finding solution to engineering design problems. Souza et al. (2022) proposed GA with simulated annealing approach to generate schedules in deterministic and stochastic machine unavailability restrictions. As a result, the authors demonstrated the significance of GA in their research. Researchers are also researching other combinatorial optimization strategies to solve and improve solutions to optimization problems. Kundu T., & Garg H. (2021) combined Harris hawks optimization and teaching-learning-based optimization algorithm and found to be better to find solution to numerical optimization problems.

The paper is further divided into the sections listed below. Section 2 discusses methodology, including performance measures, deterministic and stochastic scheduling, and their approaches. Section 3 describes the optimization of the non-tardiness and tardiness-related measures, as well as the GA algorithm. Section 4 includes a case study of a manufacturing company as well as the model developed for it. Section 5 discusses the case study results. Finally, section 6 shows the conclusions of the article.

## 2. Methodology

The main objective of this article is to develop a model to find the optimal solution to scheduling problems using GA. The model was created using the Python programming language on a computer with 8 GB of RAM, a 500 GB hard drive, and the Windows 10 operating system. This model aids in determining the best sequence for minimizing performance measures in accordance with industry standards. For deterministic, stochastic linguistic, stochastic probabilistic, and stochastic scenarios-based scheduling problems, the model finds the best solution. The following are the scheduling performance measures:

1. Completion time ( $C_j$ ): It is the period necessary to finish a single job  $j$ . Consider five jobs,  $j_1, j_2, j_3, j_4$  and  $j_5$ , with their respective PT and DD as  $p_1, p_2, p_3, p_4, p_5$  and  $d_1, d_2, d_3, d_4, d_5$ . Additionally, job's In-Time and Out-Time, that is, the time at which the processing operation of a specific job, from a list of jobs, begins and ends, respectively, is also required. Table 1 shows the completion time of each job under the Out-Time column. Çetinkaya & Duman, (2021) developed an approach to minimize completion time of the sublots and job lots with a single job and multiple jobs.

Table 1. Data of 5 jobs

Job Number ( $j$ )	PT ( $p_j$ )	In-Time	Out-Time ( $C_j$ )	DD ( $d_j$ )
$j_1$	$p_1$	0	$p_1$	$d_1$
$j_2$	$p_2$	$p_1$	$p_1 + p_2$	$d_2$
$j_3$	$p_3$	$p_1 + p_2$	$p_1 + p_2 + p_3$	$d_3$
$j_4$	$p_4$	$p_1 + p_2 + p_3$	$p_1 + p_2 + p_3 + p_4$	$d_4$
$j_5$	$p_5$	$p_1 + p_2 + p_3 + p_4$	$p_1 + p_2 + p_3 + p_4 + p_5$	$d_5$

2. Flow time ( $F_j$ ): It is the period between the finishing time of a job and the starting time, that is, the difference between the Out-Time and release time of the job. For the deterministic model,  $r_j = 0$ , where  $r_j$  is the job's release time or starting time. The relationship is defined by equation 1.

$$F_j = C_j - r_j \quad (1)$$

3. Total flow time ( $TFT$ ): It is the cumulative flow time for all the jobs and represented by equation 2. From Table 1,  $TFT$  is the summation of the values in the Out-Time column.

$$TFT = \sum_{j=1}^n F_j \quad (2)$$

4. Average job completion time ( $T_{avg}$ ): It is the ratio of  $TFT$  to the number of jobs ( $n$ ) in a given set and shown in equation 3.

$$T_{avg} = \frac{TFT}{n} \quad (3)$$

5. Average number of jobs in the system ( $N_{avg}$ ): It is the ratio of  $TFT$  and the job with the maximum flow time, i.e.  $\max(F_j)$  formulated as equation 4.

$$N_{avg} = \frac{TFT}{\max(F_j)} \quad (4)$$

6. Percentage utilization (% Utilization): It is the reciprocal of the average number of jobs in the system given in equation 5. It is defined as the number of machines available in a job shop used to process a job.

$$\% \text{ Utilization} = \frac{\max(F_j)}{TFT} \quad (5)$$

7. Lateness ( $L_j$ ): It is the difference between the completion time and the DD of a job expressed in equation 6. From Table 1, the lateness of a job is the difference between the Out-Time and corresponding DD. It can have either a positive, negative or zero value. If lateness is positive, the job will be delayed, whereas, if it is negative, the job will be completed before the DD. If lateness is zero, the job will be completed on time.

$$L_j = C_j - d_j \quad (6)$$

8. Average lateness ( $L_{avg}$ ): It is the ratio of the lateness of all the jobs in the system to the number of jobs shown in equation 7.

$$L_{avg} = \frac{1}{n} \sum_{j=1}^n L_j \quad (7)$$

9. Tardiness ( $T_j$ ): It is the measure of the delay in the completion of a job beyond the DD formulated as equation 8. Tardiness can have either a positive or zero value. If the difference between completion time and DD is negative, the job is early and not tardy; hence, the tardiness in such a case will be 0.

$$T_j = \max(0, C_j - d_j) \quad (8)$$

10. Total tardiness ( $T$ ): It is the cumulative delay of all the jobs in the set represented in equation 9. It is the summation of the tardiness of all jobs.

$$T = \sum_{j=1}^n T_j \quad (9)$$

11. Maximum tardiness ( $T_{max}$ ): It is the measure of the maximum delay of a job beyond the DD.

12. Average tardiness ( $T_{avg}$ ): It is the ratio of total tardiness and the number of jobs in the system shown in equation 10.

$$T_{avg} = \frac{T}{n} \quad (10)$$

13. Number of tardy jobs ( $N_{tj}$ ): It is a measure of the number of delayed jobs in the system and is expressed in equation 11.

$$N_{tj} = \sum_{j=1}^n \delta(T_j) \quad \begin{cases} \delta(x) = 1 \text{ if } x > 0 \\ \delta(x) = 0 \text{ otherwise} \end{cases} \quad (11)$$

## 2.1. Deterministic Scheduling

The deterministic scheduling operation is done considering only the present scenario at hand. This type of scheduling requires only the jobs' PT and DD. The DD given by the client remains fixed. The PT changes depending on the nature of the factors affecting the jobs. Deterministic scheduling does not take into account these factors. Hence, deterministic scheduling can be referred to as an idealistic operation. The deterministic model considers assumptions like jobs are available simultaneously for processing, a machine can process only one job at a time, set up times are included in the PT, input data is deterministic and known in advance, machines are available

continuously, and never kept idle, once an operation begins on the machine it proceeds without interruption (French, 1982). An illustration of the data required for deterministic scheduling is given in Table 2. Consider five jobs,  $j_1, j_2, j_3, j_4$  and  $j_5$ , with their respective PT and DD as  $p_1, p_2, p_3, p_4, p_5$  and  $d_1, d_2, d_3, d_4, d_5$ .

*Table 2. Data format for deterministic scheduling*

Job Number ( $j$ )	PT ( $p_j$ )	DD ( $d_j$ )
$j_1$	$p_1$	$d_1$
$j_2$	$p_2$	$d_2$
$j_3$	$p_3$	$d_3$
$j_4$	$p_4$	$d_4$
$j_5$	$p_5$	$d_5$

## 2.2. Stochastic Scheduling

In real-world scheduling problems, various factors affect the PT of jobs and cannot be neglected. The deviation that occurred in the accuracy of the PT values can hamper the efficacy of the job shop. This further can cause a delay in circumstances where these factors are in the worst case. To assist the industry in scheduling of jobs with real-time data, stochastic scheduling is preferred, which provides more accurate results. This also gives an idea to the operators about when and how to sequence the jobs and provides the client with a view to set the DD of procurement. The assumptions like input data is deterministic and known in advance, and machines are available continuously and never kept idle made in deterministic scheduling are relaxed in stochastic scheduling. Through the literature review, three stochastic scheduling models, as mentioned below (Baker & Trietsch, 2009), have been identified that can provide the required results.

### 2.2.1. Scenarios Method

In this method, scheduling is done considering more than one scenario of the jobs as mentioned in Table 3. These scenarios are for the same jobs; that is, the end product is the same. However, the path to preparing the end product in different scenarios may be different. For example, in some scenarios, jobs can be made of different materials or the size of the raw material can be different, or the machine used to process can be different. These various factors affecting the jobs can increase or decrease the PT of the jobs. Thus, different scenarios are created to assist in stochastic scheduling with real-time data.

Consider five jobs,  $j_1, j_2, j_3, j_4$ , and  $j_5$  with the respective PT of  $p^1_1, p^1_2, p^1_3, p^1_4, p^1_5$ , (for scenario 1)  $p^2_1, p^2_2, p^2_3, p^2_4, p^2_5$ , (for scenario 2) and  $p^3_1, p^3_2, p^3_3, p^3_4, p^3_5$ . (for scenario 3). The DD for the five jobs are  $d_1, d_2, d_3, d_4$ , and  $d_5$ . This data is shown in Table 3.

*Table 3. Data format for scenarios scheduling*

Job Number	Scenario 1 PT	Scenario 2 PT	Scenario 3 PT	DD
$j_1$	$p^1_1$	$p^2_1$	$p^3_1$	$d_1$
$j_2$	$p^1_2$	$p^2_2$	$p^3_2$	$d_2$
$j_3$	$p^1_3$	$p^2_3$	$p^3_3$	$d_3$
$j_4$	$p^1_4$	$p^2_4$	$p^3_4$	$d_4$
$j_5$	$p^1_5$	$p^2_5$	$p^3_5$	$d_5$

In deterministic scheduling, only the PT of one scenario is considered at a given time. The scenario can be 1, 2 or 3 from Table 3. However, in the stochastic model, the PT of every job is considered by taking the average of the PT from all the possible scenarios that can occur and shown in equation 12. Thus, the PT for any job concerning the above table can be given as:

$$\text{Expected PT of } j^{\text{th}} \text{ job} = p_j = \frac{p_j^1 + p_j^2 + p_j^3}{3} = \frac{1}{3} \sum_{i=1}^3 p_j^i$$

Hence, the formula can be given as:

$$p_j = \frac{1}{n_s} \sum_{j=1}^{n_s} p_j \tag{12}$$

where,  $n_s$  – Number of scenarios and  $j$  – Scenario number

### 2.2.2. Linguistic method

The linguistic method stands out when selecting real-time data of jobs. In this method, the PT of jobs is estimated by considering various factors that affect their operation. For instance, to perform a turning operation on a job, some of the factors affecting are the condition of the tool, the machine, the material of the job or coolant type. These and many other factors can either increase or decrease the PT. If a particular factor increases the PT of a job, it may not be in its best form. For example, an increase in PT concerning the above factors can be because the tool is blunt, the machine is not operating as it should, the material of the job is rough, or the coolant is not effective enough.

Conversely, if the same factors are in their best form, the PT of the job will decrease. In the linguistic method, either one or all the factors can be in their best or worst shape under different circumstances. Thus, if there are ' $n_f$ ' factors affecting the PT of a job and each factor can have either Good (G) or Bad (B) form, the number of conditions is given by  $2^{n_f}$ . In a more general aspect, the number of conditions can be given by  $N_c = x^{n_f}$  where, ' $n_f$ ' is the number of factors and  $x$  is the number of forms each factor can have.

Consider there are five jobs, and three factors are taken into account that can affect the processing of each job. Each factor can have two forms that is G or B. Hence, there will be eight ( $2^3$ ) conditions for each job like BBB, which means all factors for jobs are in bad conditions, BBG means the first two factors are in bad conditions whereas the third factor is in good condition and so on. Table 4 shows the data format of the linguistic method with eight conditions for each job.

Table 4. Data format for linguistic scheduling

Job Number	BBB	BBG	BGB	BGG	GBB	GBG	GGB	GGG	DD
1	32	28	27	25	22	20	16	15	23
2	30	26	23	22	19	17	14	13	21
3	33	29	25	24	20	19	16	11	17
4	28	27	26	21	17	16	13	9	13
5	35	28	25	20	16	15	13	10	15

From the table, the real-time data of the jobs can be selected by the form of each factor. For example, if the condition chosen for Job Number 1, 2, 3, 4 and 5 is BGB, GBB, BBB, GBG and GGG, respectively, the new data for the scheduling operation will be shown in Table 5.

Table 5. Stochastic Scenarios Scheduling data after operator selection

Job Number ( $j$ )	PT ( $p_j$ )	DD ( $d_j$ )
1	27	23
2	19	21
3	33	17
4	16	13
5	10	15

Thus, the PT of the jobs is now based on the form of the individual factors in the present scenario.

### 2.2.3. Probabilistic Method

The probabilistic scheduling method is similar to the scenarios method, wherein each job has a different PT in each scenario. However, this method does not take an average of the PT; instead, a probability of occurring is associated with each scenario. Therefore, the sum of the probabilities of each scenario happening is equal to 1. The probabilistic approach is used for repetitive jobs carried out in industry. Ideally these jobs should have the same PT, but this PT can be different for different scenarios and thus the probability of occurring in the scenarios may vary due to different elements. The probability of scenarios is determined by elements such as machine breakdown, power failure, worker absenteeism, technology failure, material shortages, unavoidable delays, and so on. The decision-maker computes these probabilities by analyzing historical data. While calculating the performance measure, each scenario has to be arranged in the sequence, and the performance value is then calculated for the individual scenarios. Next, the expected value of the performance measure is calculated by taking the summation of the product of individual scenarios' performance measure and the probability of the scenario. Consider the data of 5 jobs,  $j_1, j_2, j_3, j_4,$  and  $j_5$ , as shown in Table 6, with the respective PT for Scenario 1 are  $p^1_1, p^1_2, p^1_3, p^1_4, p^1_5$  for Scenario 2 are  $p^2_1, p^2_2, p^2_3, p^2_4, p^2_5$  and for Scenario 3 are  $p^3_1, p^3_2, p^3_3, p^3_4, p^3_5$ . The DD for the five jobs are  $d_1, d_2, d_3, d_4,$  and  $d_5$ . The probability of Scenario 1, 2 and 3 occurring is  $P_1, P_2$  and  $P_3$ , respectively.

Table 6. Data format for Stochastic Probabilistic Scheduling

Job Number	Scenario 1 PT	Scenario 2 PT	Scenario 3 PT	DD
$j_1$	$p^1_1$	$p^2_1$	$p^3_1$	$d_1$
$j_2$	$p^1_2$	$p^2_2$	$p^3_2$	$d_2$
$j_3$	$p^1_3$	$p^2_3$	$p^3_3$	$d_3$
$j_4$	$p^1_4$	$p^2_4$	$p^3_4$	$d_4$
$j_5$	$p^1_5$	$p^2_5$	$p^3_5$	$d_5$

If the performance measure,  $TFT$  calculated for the above 3 Scenarios is  $TFT_1, TFT_2$  and  $TFT_3$ , then the *Expected TFT* is given in equation 13,

$$Expected\ TFT = TFT_1 * P_1 + TFT_2 * P_2 + TFT_3 * P_3 \quad (13)$$

The same approach can be implemented for finding the expected values of all the performance measures.

### 3. Optimizing Performance Measures

The performance measures are divided into two categories, non-tardiness performance measures and tardiness performance measure. The non-tardiness performance measures are optimized with the SPT rule, while for tardiness related measures, GA is applied to generate an optimal sequence.

#### 3.1. Non-Tardiness Performance Measures

Non-tardiness performance measures are total flow time, average flow time, total PT, percentage utilization, the average number of jobs in the system, total lateness and average lateness. All these measures, except percentage utilization, can be reduced by arranging the jobs in the SPT sequence. If the jobs are arranged in an SPT sequence, percentage utilization will have a maximum value. The explanation for this approach has been given using the following example.

Consider five jobs  $j_1, j_2, j_3, j_4$  and  $j_5$  with PT  $p_1, p_2, p_3, p_4$  and  $p_5$ . The DD of the jobs are  $d_1, d_2, d_3, d_4$  and  $d_5$ . This data is summarized in Table 2. To calculate the non-tardiness related performance measures, two additional columns must be added, In-time and Out-time of each job. In-time indicates when a job arrives on a machine, more specifically, after completing the preceding job, if any. Out-time indicates the time at which a job leaves the machine that is after completion of its PT.

##### 1. Total flow time

It is given by the sum of the values in the Out-Time column. Therefore, in this case,

$$\begin{aligned} TFT &= p_1 + p_1 + p_2 + p_1 + p_2 + p_3 + p_1 + p_2 + p_3 + p_4 + p_1 + p_2 + p_3 + p_4 + p_5 \\ TFT &= 5p_1 + 4p_2 + 3p_3 + 2p_4 + p_5 \end{aligned} \quad (14)$$

To minimize the value of  $TFT$  shown in equation 14, the job with the least PT has to be multiplied with the largest coefficient and so on from left to right in increasing order of PT.

##### 2. Average job completion time

$$T_{avg} = \frac{5p_1 + 4p_2 + 3p_3 + 2p_4 + p_5}{n} \quad (15)$$

To reduce  $T_{avg}$ , shown in equation 15, the number of jobs is constant, and hence the sequence which will reduce the  $TFT$  will also reduce  $T_{avg}$ .

##### 3. Percentage utilization

$$\% \text{ Utilization} = \frac{\max(\text{Flow Time})}{TFT} * 100$$

The performance measure shown in equation 16 requires to be maximum to make the utmost utilization of the machine for the given set of jobs. For example, from Table 1, job number 5 will have the maximum flow time, and the TFT must be minimized to maximize the performance measure.

$$\% \text{ Utilization} = \frac{p_1 + p_2 + p_3 + p_4 + p_5}{5p_1 + 4p_2 + 3p_3 + 2p_4 + p_5} * 100 \quad (16)$$

#### 4. Average number of jobs in the system

$$N_{avg} = \frac{1}{\% \text{ Utilization}} \quad (17)$$

To minimize this performance measure shown in equation 17, the percentage utilization performance measure must be maximum. Hence, this value can be minimized using the same approach from the previous performance measure.

#### 5. Total lateness

Total lateness is the sum of the difference between the Out-time of a job and its DD. From the Table 1, the equation obtained is,

$$\text{Total lateness} = (p_1 - d_1) + (p_1 + p_2 - d_2) + (p_1 + p_2 + p_3 - d_3) + (p_1 + p_2 + p_3 + p_4 - d_4) + (p_1 + p_2 + p_3 + p_4 + p_5 - d_5)$$

$$\text{Total lateness} = (5p_1 + 4p_2 + 3p_3 + 2p_4 + p_5) - (d_1 + d_2 + d_3 + d_4 + d_5)$$

$$\text{Total Lateness} = TFT - (d_1 + d_2 + d_3 + d_4 + d_5) \quad (18)$$

The summation of the DD is the same irrespective of the job sequence. Hence, the TFT must be minimum to reduce the lateness shown in equation 18. Therefore, a minimum value can be obtained using the same sequence of jobs from the previous measures.

#### 6. Average lateness

The average lateness is shown in equation 19. The number of jobs is constant. So, to minimize the numerator, the same approach from the previous performance measure can be used.

$$L_{avg} = \frac{\text{Total Lateness}}{n} \quad (19)$$

#### 7. Total PT

It is the summation of the PT of all jobs in the scheduling problem. The total PT is a constant value for any given sequence of jobs and cannot be minimized.

### 3.2. Tardiness Performance Measures

For the tardiness related parameters, a few of the approaches tested to minimize the performance measures were the Branch and Bound algorithm (Tyagi et al., 2016), Excel Workbook tool, sequentially searching through each of the ' $n!$ ' sequences possible. However, these approaches were limited by various factors, including computing time and data size variation. Hence, a randomized search approach was implemented like GA (Bancila & Buzatu, 2008).

The GA initiates by using random sequences of the jobs as starting population. Then, it proceeds with each sequence of the population over a fitness function. Later, it chooses the fittest sequence of the population to reproduce using the reproduction function of GA and repeats the assessment and reproduction until a selected number of iterations. In the end, the algorithm presents the optimal sequence of the population according to the fitness function. Following are the steps in the GA used in the simulation model:

#### *Step 1: Select the initial population randomly*

The initial population is a set of randomly generated chromosomes (sequence of jobs) as input to the GA. GA chooses a set of samples randomly from  $n!$  sequences as the initial population.

For example, the 30 sequences are selected randomly for five jobs 0,1,2,3,4 from 5! that is 120 sequences as an initial population shown below.

[2, 0, 1, 4, 3]	[1, 4, 2, 0, 3]	[4, 1, 2, 3, 0]	[3, 2, 0, 4, 1]	[1, 2, 4, 0, 3]
[3, 4, 1, 0, 2]	[0, 4, 2, 1, 3]	[2, 4, 1, 3, 0]	[4, 1, 0, 3, 2]	[3, 0, 2, 4, 1]
[0, 1, 2, 3, 4]	[1, 3, 4, 2, 0]	[4, 2, 1, 0, 3]	[4, 2, 0, 3, 1]	[1, 3, 4, 2, 0]
[4, 3, 2, 1, 0]	[4, 0, 2, 3, 1]	[4, 3, 0, 2, 1]	[0, 3, 2, 1, 4]	[4, 1, 3, 0, 2]
[2, 4, 1, 3, 0]	[1, 3, 2, 4, 0]	[4, 1, 0, 3, 2]	[2, 1, 4, 0, 3]	[2, 1, 3, 4, 0]
[0, 1, 4, 2, 3]	[4, 0, 3, 1, 2]	[2, 4, 1, 3, 0]	[3, 0, 4, 1, 2]	[4, 3, 2, 1, 0]

#### *Step 2: Application of GA operators*

Crossover operator: Crossover is a genetic operator used to modify a chromosome or chromosomes by combining chromosomes from one generation to the next. Randomly select two parents, just for manageable iteration, number all the sequences and shuffle it for reproduction operation on the population to get offspring.

0	6	12	18	24
1	7	13	19	25
2	8	14	20	26
3	9	15	21	27
4	10	16	22	28
5	11	17	23	29

[26, 7, 9, 29, 3, 10, 2, 28, 8, 27, 21, 16, 25, 12, 20, 15, 4, 24, 6, 19, 5, 18, 14, 0, 13, 23, 11, 22, 17, 1]

Let's select the first two sequences as parents 26 and 7.

P1 [1, 3, 4, 2, 0]

P2 [0, 4, 2, 1, 3]

Randomly select half of the number of jobs and create the sequences just by swapping the jobs at that positions to get children. Now remaining places are filled with jobs that are not in children.

Let's select [0,4]

Child\_1 = [0, 'na', 'na', 'na', 3]

Child\_2 = [1, 'na', 'na', 'na', 0]

'na' represents the blank place

Thus in Child\_1, jobs [1, 4, 2] are not present. Similarly, in Child\_2, jobs [4, 2, 3] are not present. Now just put these jobs to get Child\_1 and Child\_2 as follows

Child\_1 = [0,1,4,2,3]

Child\_2 = [1,4,2,3,0]

This process needs to repeat for the remaining sequences, producing the different sequences as offspring list.

*Mutation:* Mutation operators are generally viewed as a random disturbance term of the individual chromosome. The children list of sequences produced by the crossover function of GA is used as input to this mutation function. First, randomly select the sequence and swap the position of jobs that gives the new list of offspring sequences.

Let's take Child\_2 = [1,4,2,3,0]

Select any two positions and swap the jobs. For example, let's select 2<sup>nd</sup> and 3<sup>rd</sup> positions and change the jobs to produce a new sequence [1,2,4,3,0]. This process repeats for all sequences present in the children list, input for mutation function to get new offspring.

#### *Step3: Evaluation of offspring*

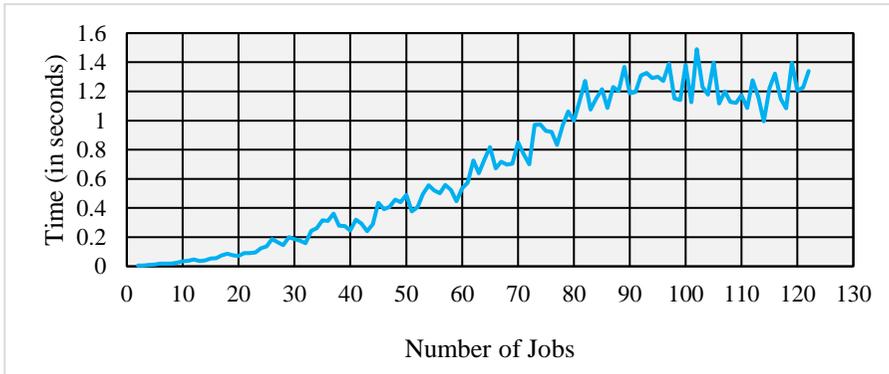
Evaluation of offspring is finding the sequence with the lowest tardiness value. Now parent list and offspring list are merged to get the whole list of sequences to find optimal sequences. The tardiness performance measures are needed to minimize. The total tardiness of all the sequences is calculated, and the sequence with minimum tardiness is selected as the optimal sequence. The exact sequence is used to calculate other tardiness based measures.

#### *Step 4: Termination Condition*

The termination condition is the stopping criterion for the algorithm. In this paper, the user gives the number of iterations as a termination condition for finding the best sequence. After a specified iteration, the algorithm stops and produces the optimal sequence for minimizing total tardiness.

## Simulation of Job Sequencing for Stochastic Scheduling with a Genetic Algorithm

Based on the GA implemented, Figure 1 shows a graph that indicates the time required to compute total tardiness, average tardiness, number of tardy jobs and maximum tardiness in one iteration, against the number of jobs in the dataset.



*Figure 1. Number of jobs and time required for 1 iteration*

From Figure 1, it is observed that the time required to compute one iteration roughly increases as the number of jobs increases. This gives the operator an idea of how much time will be required for computing the required number of iterations. Another benefit of the implemented algorithm is that if the values reach a minimum value, the algorithm will check the value for a few more iterations and exit the loop after a few iterations to save the operator's time from computing the remaining number of iterations. Hence implementing the SPT approach and a GA, an optimizing stochastic model can be developed for any scheduling application.

## 4. Case Study

A case study was performed in association with a manufacturing company specializing in pipe fittings and flanges to verify the methodology and compare results with the real-world scenario. Figure 2 shows sample data from a datasheet of the company. The model developed for the case study uses the scenarios method of scheduling explained earlier in section 2.2.1. The model is limited to  $n$  jobs one machine scenario that is scheduling will be done for any number of jobs as long as they are processed on one machine. With respect to Figure 2, the simulation model assigns job numbers based on the sizes of the individual components; that is, job numbers are assigned based on the 'Size' column. The model also calculates the PT of the respective job, in minutes, by taking the product of the values from the column 'Qty.' and 'Processing time'. The PT is then converted to hours because the DD of the jobs is also accepted in terms of hours. Thus, the data required for performing scheduling operations is prepared. The model is created using Python programming language on a computer system with 8 GB RAM and 500 GB hard disk and Windows 10 operating system. This model can also run on other versions of the operating system like Windows 7, Windows 8 and other operating systems. Other than this, no other special hardware and software requirements are needed.

Item	Specification	Size (NB)	Qty.	Processing Time	Process Used
Elbows 90°	CS, 90° elbow, R=1.5D, Socket weld end, dimensions as per ANSI B 16.11 3000# rating, material as per ASTM A 105	15	137	6 min/ 1 no	Socket end weld turret lathe
		20	58	6 min/ 1 no	Socket end weld turret lathe
		25	438	6 min/ 1 no	Socket end weld turret lathe
	CS, 90° elbow, R=1.5D, Bevelled End, dimensions as per ANSI B 16.9 Sch. 40, material shall be wrought steel A 234 Gr. WPB, Seamless.	32	233	6 min/ 1 no	Beveled end weld turret lathe
		40	377	6 min/ 1 no	Beveled end weld turret lathe
		50	173	6 min/ 1 no	Beveled end weld turret lathe
		65	128	6 min/ 1 no	Beveled end weld turret lathe
		80	522	6 min/ 1 no	Beveled end weld turret lathe
		100	465	6 min/ 1 no	Beveled end weld turret lathe
	CS, 90° elbow, R=1.5D Bevelled End, dimensions as per ANSI B 16.9 Sch. 10., material shall be wrought steel A 234 Gr. WPB, Seamless.	125	402	6 min/ 1 no	Beveled end weld turret lathe
		150	234	6 min/ 1 no	Beveled end weld turret lathe
		15	76	6 min/ 1 no	Beveled end weld turret lathe
		20	48	6 min/ 1 no	Beveled end weld turret lathe
Elbows 45°	Specs same as above but 45°	25	56	6 min/ 1 no	Beveled end weld turret lathe
		32	52	6 min/ 1 no	Beveled end weld turret lathe
		40	49	6 min/ 1 no	Beveled end weld turret lathe
		50	43	6 min/ 1 no	Beveled end weld turret lathe
		65	38	6 min/ 1 no	Beveled end weld turret lathe
		80	29	6 min/ 1 no	Beveled end weld turret lathe
		100	29	6 min/ 1 no	Beveled end weld turret lathe
		125	28	6 min/ 1 no	Beveled end weld turret lathe
		150	19	6 min/ 1 no	Beveled end weld turret lathe

Figure 2. Sample data of case study

A similar approach is followed if there are more than one scenario. If there are more than one scenario, the simulation model requires the same number and similar jobs in all the datasheets. Figure 3 shows the Main Window of the job shop scheduling simulator. All four scheduling methods explained earlier are included in this simulator. A new window will open for the respective scheduling model on clicking any button. The 'About' button provides information about the simulator. For example, the case study model developed can be located by clicking the 'Stochastic Scheduling (Scenarios) Company Model' button. On clicking the company model button, the main window of the model will open up, as shown in Figure 4. The model has two ways of data input, manual input or importing an excel file. The 'Enter Number of Iterations' box is required only for the tardiness related performance measures, without which error will be displayed. If there are more than one scenario, the operator can select the individual scenarios by entering in the 'Enter Scenarios of Jobs' box. The 'Instructions' give information about the file format and the format of input wherever required.

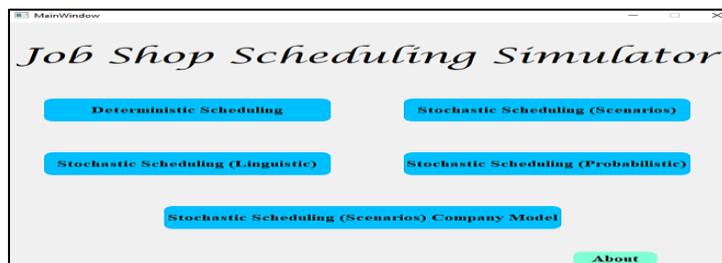


Figure 3. Main Window of Simulator

## Simulation of Job Sequencing for Stochastic Scheduling with a Genetic Algorithm

MainWindow

### Job Shop Scheduler (Scenarios Model)

Import Data Files:  Manual Entry:

Enter Number of Iterations:

There are 0 Scenarios in the given data.

Enter Scenarios of Jobs:

Performance Measures:

Total Flow Time  Total Processing Time  Total Tardiness

Average Number of Jobs in the System  Average Tardiness  Number of Tardy Jobs

Average Job Completion Time  Percentage Utilization  Average Lateness

Maximum Tardiness  Total Lateness  All Performance Measures

Figure 4. Main Window of Company Model

If the operator chooses to import the data, a 'Browse' window will appear to select one or more files from the device storage. An error will be displayed if the datasheet is not in the proper format. After selecting the file(s), the software will read through the 'Process used' column of the datasheet(s) to identify different machines. Now the operator has to enter for which machine scheduling has to be performed, as the simulation will be done for  $n$  jobs one machine. This input will be taken in the 'Select Process' window, as shown in Figure 5. The operator can also enter the DD of the jobs, in hours, in the same window. On clicking the 'Submit' button, the data will be ready for scheduling operation.

Select Process

Enter Process:

Processes Found: LATHE, VMC

Enter Common Due Date:

Figure 5. Process used and due date

The operator can also enter the data of the jobs manually, that is, the DD and PT of the jobs, by clicking the 'Enter Data' button in the main window. On clicking, a window, as shown in Figure 6, will appear. The simulator will automatically assign the job number in the order the input is given. The PT of more than one scenario can also be added there. The units, however, of both the inputs have to be the same as no units are assumed or assigned here.

Enter Due Date:

Enter Processing Time for Scenarios:

For the jobs in the same scenario, separate the Processing Time by a comma ',';  
 For separating different scenarios use a semi-colon ',';  
 Separate the Due Date for each job by a comma ','

*Figure 6. Window to manually enter data*

The operator must select the performance measures for which the minimizing sequence has to be found and the corresponding minimum value. The individual performance measures can be chosen, or the 'All Performance Measures' can be selected as a whole. If the operator selects 'All Performance Measures' or any of the tardiness related performance measures, input will be required from the operator for the number of iterations. The computing time will increase if the number of iterations are more or if the number of jobs are more. The operator can give the appropriate value based on the time at hand. The operator is required to enter the scenarios of the jobs. This can be done in the following ways.

1. For selecting single scenarios, the scenario number can be given, that is, 1 or 2
2. For more than one scenario, each scenario should be separated with a comma, 3, 4, 5, 6, 7.
3. The hyphen symbol can be used for selecting a range of scenarios; that is, 4-10 will select all the scenarios within this range, including 4 and 10.
4. For a combination of individuals and a range of scenarios, the following format can be followed that is, 4-10, 12, 13, 16. This will select the scenarios from the range of 4 and 10, including the individual scenarios of 12, 13, and 16.

The operator can also view the data table, that is, the job number, PT and DD of every job and scenario, by clicking on the 'Preview Table' button. On clicking, a window, as shown in Figure 7, will appear. The data, here, cannot be manipulated. After all the required input is given for the scheduling operation, as shown in Figure 8, the scheduling operation is ready to be performed.

## Simulation of Job Sequencing for Stochastic Scheduling with a Genetic Algorithm

Job Number	Scenario 1 Processing Time	Scenario 2 Processing Time	Due Date
1.0	2.9	3.867	700.0
2.0	1.9	2.533	700.0
3.0	2.8	3.733	700.0
4.0	2.9	3.867	700.0
5.0	3.8	5.067	700.0
6.0	4.3	5.733	700.0
7.0	4.9	6.533	700.0
8.0	5.2	6.933	700.0
9.0	5.6	7.467	700.0
10.0	4.8	6.4	700.0
11.0	7.6	10.133	700.0
12.0	13.7	20.55	700.0
13.0	40.2	60.3	700.0
14.0	46.5	69.75	700.0
15.0	52.2	78.3	700.0
16.0	12.8	19.2	700.0
17.0	17.3	25.95	700.0
18.0	37.7	56.55	700.0
19.0	23.3	34.95	700.0
20.0	43.8	65.7	700.0

Figure 7. Sample Preview Table Window

*Job Shop Scheduler (Scenarios Model)*

Import Data Files:       Manual Entry:

Enter Number of Iterations:

There are 2 Scenarios in the given data.

Enter Scenarios of Jobs:

Performance Measures:

Total Flow Time     
  Total Processing Time     
  Total Tardiness  
 Average Number of Jobs in the System     
  Average Tardiness     
  Number of Tardy Jobs  
 Average Job Completion Time     
  Percentage Utilization     
  Average Lateness  
 Maximum Tardiness     
  Total Lateness     
  All Performance Measures

Figure 8. Performance measure selection

A new window will appear after clicking the 'Submit' button, as shown in Figure 9. This window will consist of all the names of the performance values selected by the operator and the corresponding minimum value in the top table. Next, in the bottom table, the job details will be provided. Finally, in the right-hand side section of the window, the names of the performance measures will be given and the sequence that will minimize performance measures. Thus, the result window consists of all the data required to perform the scheduling operation.

Performance Measure		Value	To maximize Percentage Utilization and minimize Total Flow Time, Average Job Completion Time, Average Number of Jobs in the System, Total Processing Time, Total Lateness, Average Lateness follow the sequence: 44, 87, 77, 55, 54, 41, 73, 59, 58, 72, 76, 45, 40, 61, 74, 65, 57, 78, 38, 70, 60, 75, 84, 68, 62, 63, 69, 89, 46, 85, 24, 66, 67, 86, 81, 48, 71, 80, 88, 64, 43, 25, 47, 79, 35, 23, 56, 42, 90, 52, 50, 49, 51, 53, 39, 2, 36, 29, 3, 4, 1, 82, 33, 31, 30, 28, 32, 34, 83, 5, 27, 6, 37, 10, 7, 26, 8, 9, 21, 11, 16, 12, 17, 19, 22, 18, 13, 20, 14, 15.  To minimize Maximum Tardiness, Total Tardiness, Average Tardiness, Number of Tardy Jobs follow the sequence: 75, 13, 2, 82, 51, 77, 55, 27, 72, 63, 47, 74, 21, 33, 48, 1, 78, 32, 42, 12, 53, 81, 56, 54, 43, 5, 79, 14, 28, 46, 50, 66, 45, 3, 10, 90, 6, 87, 57, 65, 31, 71, 52, 15, 7, 8, 49, 60, 9, 29, 69, 41, 25, 4, 86, 44, 34, 83, 17, 80, 68, 36, 67, 84, 89, 30, 23, 16, 18, 26, 37, 76, 70, 59, 62, 19, 38, 61, 88, 73, 35, 24, 58, 20, 22, 64, 11, 40, 39, 85.
1	Total Flow Time	6528.85	
2	Average Job Completion Time	72.543	
3	Average Number of Jobs in the System	11.923	
4	Percentage Utilization	8.387	
5	Total Processing Time	547.6	
6	Total Lateness	-56471.15	
7	Average Lateness	-627.457	
8	Maximum Tardiness	0	
9	Total Tardiness	0	
10	Average Tardiness	0.0	
11	Number of Tardy Jobs	0	

Job	Average Processing Time	Due Date
1 Elbows 45° (80)	3.38	700.0
2 Elbows 45° (150)	2.22	700.0
3 Elbows 45° (125)	3.27	700.0
4 Elbows 45° (100)	3.38	700.0
5 Elbows 45° (65)	4.43	700.0
6 Elbows 45° (50)	5.02	700.0
7 Elbows 45° (40)	5.72	700.0

Figure 9. Results for due date 700

The DD given by the client was four months, that is, 700 hours (assuming 25 working days in a month and seven working hours per day). After running the model with DD of 700 hours, the tardiness comes to be 0, as shown in Figure 9. This indicates that all the jobs are completed early. The just-in-time approach in manufacturing industries stimulates the notion of tardiness and earliness also. In industry, in view of due-dates, the primary intent is to complete all the jobs on time. The intent may be achieved by permitting the loose DD as 700 hours as DD in the case study. In this case of unrestricted DD situation, all jobs can be completed before DD by any sequence. Completing the jobs before time affects inventory carrying costs like storage and protection costs. However, due dates should be chosen carefully, as DD that is tight or restricted invite more customers. The above discussion indicates that the DD should be tighter. If the DD is negotiable with the customers and can be made still tighter, it will attract more customers, and there will be no need to keep a loose DD. After several runs with a different DD, it is found that the DD of 548 hours, approximately 3 months and 3 days is optimum for the case study. The DD produces the same result for tardiness measures, as shown in Figure 10. It is noted that the sequence of jobs for non-tardiness performance measures is the same as it is based on the SPT priority sequencing rule; however, the sequence for tardiness performance measure is different for both cases shown in Figure 9 and Figure 10. The average lateness is reduced from 627.457 to 475.457, which indicates that the inventory level of the completed job can be minimized.

## Simulation of Job Sequencing for Stochastic Scheduling with a Genetic Algorithm

Performance Measure		Value	To maximize Percentage Utilization and minimize Total Flow Time, Average Job Completion Time, Average Number of Jobs in the System, Total Processing Time, Total Lateness, Average Lateness follow the sequence: 44, 87, 77, 55, 54, 41, 73, 59, 58, 72, 76, 45, 40, 61, 74, 65, 57, 78, 38, 70, 60, 75, 84, 68, 62, 63, 69, 89, 46, 85, 24, 66, 67, 86, 81, 48, 71, 80, 88, 64, 43, 25, 47, 79, 35, 23, 56, 42, 90, 52, 50, 49, 51, 53, 39, 2, 36, 29, 3, 4, 1, 82, 33, 31, 30, 28, 32, 34, 83, 5, 27, 6, 37, 10, 7, 26, 8, 9, 21, 11, 16, 12, 17, 19, 22, 18, 13, 20, 14, 15.  To minimize Maximum Tardiness, Total Tardiness, Average Tardiness, Number of Tardy Jobs follow the sequence: 1, 73, 83, 75, 42, 43, 4, 9, 37, 2, 29, 68, 11, 12, 52, 8, 30, 55, 63, 22, 70, 54, 38, 79, 16, 13, 50, 60, 85, 19, 47, 33, 32, 77, 61, 69, 87, 17, 21, 81, 64, 15, 62, 86, 39, 48, 36, 49, 74, 20, 72, 41, 25, 35, 89, 14, 3, 23, 18, 80, 53, 40, 67, 76, 90, 66, 24, 10, 59, 56, 88, 27, 45, 44, 31, 26, 34, 28, 84, 58, 51, 78, 46, 82, 5, 71, 65, 7, 57, 6.
1	Total Flow Time	6528.85	
2	Average Job Completion Time	72.543	
3	Average Number of Jobs in the System	11.923	
4	Percentage Utilization	8.387	
5	Total Processing Time	547.6	
6	Total Lateness	-42791.15	
7	Average Lateness	-475.457	
8	Maximum Tardiness	0	
9	Total Tardiness	0	
10	Average Tardiness	0.0	
11	Number of Tardy Jobs	0	

Job	Average Processing Time	Due Date
1 Elbows 45° (80)	3.38	548.0
2 Elbows 45° (150)	2.22	548.0
3 Elbows 45° (125)	3.27	548.0
4 Elbows 45° (100)	3.38	548.0
5 Elbows 45° (65)	4.43	548.0
6 Elbows 45° (50)	5.02	548.0
7 Elbows 45° (40)	5.72	548.0

Figure 10. Results for due date 548

### 5. Results and Discussion of Case Study

The case study is conducted using the DD given by the client with a stochastic scenario model. The company sequenced the jobs in the FCFS rule. There were 90 jobs for the company to schedule on the lathe machine for sequencing. The performance measures were calculated for the dataset, with the DD as four months for the dataset. The company followed a deterministic approach to job scheduling. Hence, only one datasheet was considered, whereas the model developed uses the stochastic approach and therefore uses two datasheets for two scenarios. Table 7 shows the calculated values for the dataset and the values calculated by the model.

Table 7. Deterministic and Stochastic Results for Data Set

Sr. No.	Performance Measures	Deterministic Approach	Stochastic Approach
1.	Total Flow Time	32967.75 hours	6528.85 hours
2.	Average Job Completion Time	366.30 hours	72.543 hours
3.	Total Processing Time	547.6 hours	547.6 hours
4.	Percentage Utilization	1.442 %	8.387 %
5.	Average Number of Jobs in the	69.338	11.923
6.	Total Lateness	-30032.2 hours	-56471.15
7.	Average Lateness	-333.69 hours	-627.457 hours
8.	Total Tardiness	0	0
9.	Average Tardiness	0	0
10.	Maximum Tardiness	0	0
11.	Number of Tardy Jobs	0	0

Both the approaches assign job numbers based on the sizes and consider the quantity of those sizes as one job. There were 90 jobs in the dataset, and to calculate the tardiness performance measures, ten iterations as termination condition were given for GA. From Table 7, it is observed that the developed model, which uses SPT

and GA methods for calculating non-tardiness and tardiness performance measures, respectively, provides a better value. Due to the proposed approach, the company will have more time after completing a set, which can be used for various tasks like maintaining the machines or beginning the operations on the next set of jobs. It is further suggested that if the DD is negotiable, that means a tighter DD as compared to the given DD by the client; the company can clear out the inventory of finished products without affecting performance measures. Finishing the jobs well beforehand also reduces inventory costs in terms of storage or protection and improves the company's value. By lowering the tardiness performance measures company can lessen the delaying cost. The total tardiness for each iteration with different DD is shown in Table 8. Figure 11 shows the tardiness with respect to iteration for DD 400, 450, 500 and greater than equal to 548 hours. Table 8 and Figure 11 suggest that the DD can be tighter as 548 hours without affecting tardiness measures. It is observed that DD of less than 548 hours affects the total tardiness. Further decrease in DD increases the total tardiness. This total tardiness can be improved as the number of iterations in GA are increased. After specific iterations, it gives a constant value, but an increase in the number of iterations will increase computational time.

Table 8. Total tardiness for each iteration with different due date

Number of Iterations	DD (In hours)							
	700	650	600	550	548	500	450	400
	Tardiness (In hours)							
1	0	0	0	0	0	93	144.9	203.8
2	0	0	0	0	0	47.6	144.9	203.8
3	0	0	0	0	0	47.6	144.9	175.2
4	0	0	0	0	0	47.6	138.0	140.4
5	0	0	0	0	0	47.6	137.5	140.4
6	0	0	0	0	0	47.6	137.7	129.9
7	0	0	0	0	0	47.6	137.1	129.9
8	0	0	0	0	0	47.6	137.1	129.9
9	0	0	0	0	0	47.6	137.0	129.9
10	0	0	0	0	0	47.6	137.0	129.9

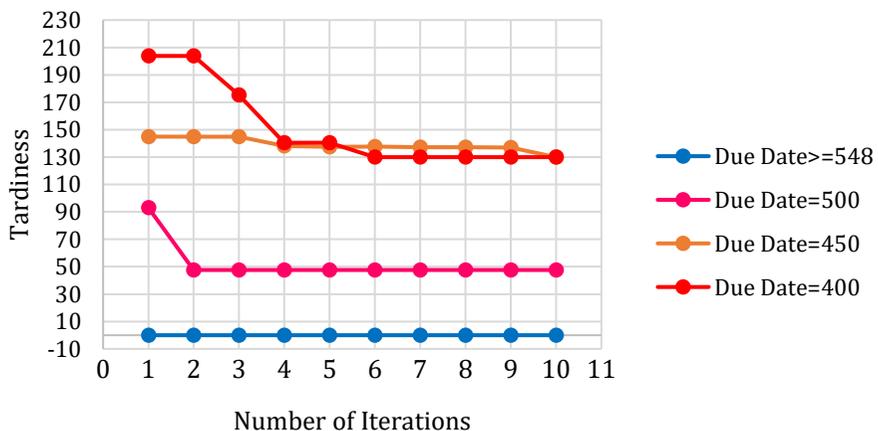


Figure 11. Total Tardiness for different due date

In comparison to traditional methods such as an excel workbook or manual calculation, the proposed work advocates using GA to reduce the computing time required to calculate performance measures and find the best sequence of jobs in a scheduling problem. In addition, the proposed work uses stochastic scenarios, linguistics, and probabilistic approaches to find job sequences that minimize performance measures. When the proposed model is applied to real data from a company, it is discovered that the sequence obtained by the model produces the lowest performance measure value when compared to the company's method. The proposed method is a unique application that allows for faster computation and better results.

## 6. Conclusions

The developed simulation model can handle the stochastic scheduling problem with linguistic, scenarios and probabilistic data to discover an optimal sequence of jobs for scheduling on a single machine. It also helps to solve the problem in a deterministic way. If the number of jobs and iterations increases, the computational time required to discover the optimal sequence also increases. When the problem was solved using the excel solver, it took more time in discovering the near-optimal sequence. The developed simulation model not only produces results with lesser time, but also improves solution. Stochastic models allow the operator to select real-time data, whereas deterministic scheduling does scheduling based on the data at hand. The SPT rule minimizes TFT and minimizes all the non-tardiness related performance measures. The developed GA model also minimizes the tardiness related performance measures. As an outcome, the proposed stochastic technique assisted the company in reducing the average completion time of job from 366.30 hours to 72.543 hours. Additionally, increased the percentage utilization from 1.442 percent to 8.387 percent. While doing the analysis of the company dataset using the developed model, it is revealed that a tighter DD is beneficial for reducing inventory costs.

The work is restricted to a single machine with an unlimited number of jobs. GA generates optimal sequences for scheduling problems, however, for different runs, it generates different sequences with the minimized performance values. To tackle this issue, a combinatorial approach of GA with other optimization techniques can be used.

Future work on this project may include developing simulation environments for  $n$  jobs  $m$  machines which will provide a broader range of options as per the industry's needs. Modifying the simulation environment from software to web application will not require instalment on the device, and the operator can use it remotely on any device. This paper focuses on stochastic scheduling and the reduction in time for completion of job. However, another factor that majorly affects scheduling is cost, which may be further analyzed. Delay of jobs can incur costs to the company of a by considerable amount. Hence, developing a simulation environment to minimize both time and cost will assist the manufacturing industries with better insight into the scheduling and sequencing of jobs.

## References

- Baker, K. R., & Trietsch, D. (2009). Principles of Sequencing and Scheduling. In *Principles of Sequencing and Scheduling*. <https://doi.org/10.1002/9780470451793>
- Bancila, D., & Buzatu, C. (2008). A hybrid algorithm to minimize the number of tardy jobs in single machine scheduling. *Annals of DAAAM and Proceedings of the International DAAAM Symposium*, 69–70. <https://doi.org/10.2507/daaam.scibook.2010.48>
- Bari, P., & Karande, P. (2020). Scheduling Problem in a Job-shop with Common Due Date to Minimize Cost and Makespan : Modeling Approach. *Manufacturing Technology and Research (An International Journal)*, 13(1–2).
- Bari, P., & Karande, P. (2021). Application of PROMETHEE-GAIA method to priority sequencing rules in a dynamic job shop for single machine. *Materials Today: Proceedings*, xxx. <https://doi.org/10.1016/j.matpr.2020.12.854>
- Bari, P., & Karande, P. (2022). Cost Minimization in a Scheduling Problem with Unrestricted and Restricted Common Due Date. *Lecture Notes in Mechanical Engineering*, 269–278. [https://doi.org/10.1007/978-981-16-5281-3\\_25](https://doi.org/10.1007/978-981-16-5281-3_25)
- Bayu, F., Panda, D., Shaik, M. A., & Ramteke, M. (2020). Scheduling of gasoline blending and distribution using graphical genetic algorithm. *Computers and Chemical Engineering*, 133, 106636. <https://doi.org/10.1016/j.compchemeng.2019.106636>
- Çetinkaya, F. C., & Duman, M. (2021). Scheduling with lot streaming in a two-machine re-entrant flow shop. *Operational Research in Engineering Sciences: Theory and Applications*, 4(3), 142–175. <https://doi.org/10.31181/ORESTA111221142C>
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017>
- French, S. (1982). *Sequencing and Scheduling: An Introduction to the Mathematics of the Job-shop* (First). Wiley.
- Garg, H. (2016). A hybrid PSO-GA algorithm for constrained optimization problems. *Applied Mathematics and Computation*, 274, 292–305. <https://doi.org/10.1016/j.amc.2015.11.001>
- Garg, H. (2019). A hybrid GSA-GA algorithm for constrained optimization problems. *Information Sciences*, 478, 499–523. <https://doi.org/10.1016/j.ins.2018.11.041>
- Koratiya, V. K., Kumar, S., & Sinha, S. (2010). Modeling, Simulation and Optimization of FCC Downer Reactor. *Petroleum and Coal*, 52(3), 183–192.
- Kumar, K. K., Nagaraju, D., Gayathri, S., & Narayanan, S. (2017). Evaluation and Selection of Best Priority Sequencing Rule in Job Shop Scheduling using Hybrid MCDM Technique. *IOP Conference Series: Materials Science and Engineering*, 197(1). <https://doi.org/10.1088/1757-899X/197/1/012059>
- Kundu T., & Garg H. (2021). A hybrid ITLHHO algorithm for numerical and engineering optimization problems. *International Journal of Intelligent Systems*, 37(3). <https://doi.org/10.1002/int.22707>
- Kurniawan, B., Chandramitasari, W., Gozali, A. A., Weng, W., & Fujimura, S. (2020). Triple-chromosome genetic algorithm for unrelated parallel machine scheduling under time-of-use tariffs. *IEEJ Transactions on Electrical and Electronic Engineering*, 15(2), 208–217. <https://doi.org/10.1002/tee.23047>

Oyetunji, E. O. (2009). Some common performance measures in scheduling problems: Review article. *Research Journal of Applied Sciences, Engineering and Technology*, 1(2), 6–9.

Ramteke, M., & Srinivasan, R. (2012). Large-scale refinery crude oil scheduling by integrating graph representation and genetic algorithm. *Industrial and Engineering Chemistry Research*, 51(14), 5256–5272. <https://doi.org/10.1021/ie201283z>

Roy, A., Manna, A., & Maity, S. (2019). A novel memetic genetic algorithm for solving traveling salesman problem based on multi-parent crossover technique. *Decision Making: Applications in Management and Engineering*, 2(2), 100–111. <https://doi.org/10.31181/dmame1902076r>

Sarkar, D., & Modak, J. M. (2005). Pareto-optimal solutions for multi-objective optimization of fed-batch bioreactors using nondominated sorting genetic algorithm. *Chemical Engineering Science*, 60(2), 481–492. <https://doi.org/10.1016/j.ces.2004.07.130>

Sharma, P., & Jain, A. (2015). Performance analysis of dispatching rules in a stochastic dynamic job shop manufacturing system with sequence-dependent setup times: Simulation approach. *CIRP Journal of Manufacturing Science and Technology*, 10, 110–119. <https://doi.org/10.1016/j.cirpj.2015.03.003>

Shrouf, F., Ordieres-Meré, J., García-Sánchez, A., & Ortega-Mier, M. (2014). Optimizing the production scheduling of a single machine to minimize total energy consumption costs. *Journal of Cleaner Production*, 67, 197–207. <https://doi.org/10.1016/j.jclepro.2013.12.024>

Souza R., Ghasemi A., Saif A., Gharaei A. (2022). Robust job-shop scheduling under deterministic and stochastic unavailability constraints due to preventive and corrective maintenance, *Computers & Industrial Engineering*, 168, 108130, ISSN 0360-8352, <https://doi.org/10.1016/j.cie.2022.108130>.

Stanković, A., Petrović, G., Čojbašić, Ž., & Marković, D. (2020). An application of metaheuristic optimization algorithms for solving the flexible job-shop scheduling problem. *Operational Research in Engineering Sciences: Theory and Applications*, 3(3), 13–28. <https://doi.org/10.31181/oresta20303013s>

Tyagi, N., Tripathi, R. P., & Chandramouli, A. B. (2016). Single machine scheduling model with total tardiness problem. *Indian Journal of Science and Technology*, 9(37), 1–14. <https://doi.org/10.17485/ijst/2016/v9i37/97527>

© 2022 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

