

# INTEGRATING PROCESS PLAN AND PART ROUTING USING OPTIMIZATION VIA SIMULATION APPROACH

Gocken, T.; Dosdogru, A. T.; Boru, A. & Gocken, M.

Industrial Engineering Department, Adana Alparslan Türkeş Science and Technology University,  
01250, Adana, Turkey

E-Mail: tgocken@atu.edu.tr, adosdogru@atu.edu.tr, aboru@atu.edu.tr, mgocken@atu.edu.tr

## Abstract

Determining the best process plan and route for each part is one of the main problems in dynamic stochastic systems. Therefore, multiple process plans are considered for each operation of each part (machine flexibility and/or part routing) and alternative operations (operation flexibility) simultaneously. In this paper, Optimization via Simulation (OvS) is utilized to plan the processes and route the parts in a dynamic stochastic flexible job-shop environment (DSFJS). Genetic algorithm (GA) which is envisaged to be the optimization component of OvS mechanism is integrated with the simulation model of the production system. A four-factor full factorial design is used to analyse the effect of main factors' and factor interactions' effects on the total of average flowtimes of each part performance of the shop. The design includes the flexibility level of the shop, number of parts, number of operations, and number of alternative process plans. Finally, the main findings of cases are summarized in the study.

(Received in November 2018, accepted in April 2019. This paper was with the authors 1 month for 1 revision.)

**Key Words:** Dynamic Stochastic Flexible Job-Shop Scheduling, Process Plan, Part Routing, Optimization via Simulation

## 1. INTRODUCTION

Flexible Job-Shop Scheduling Problem (FJSP) can be defined as the generalization of the classical Job-Shop Scheduling Problem (JSP) and the parallel machine environment. It can be divided into routing and scheduling as a two sub-problems. In routing sub-problem, each operation is assigned to a machine determined from an alternative machine set. In scheduling sub-problem, the assigned operations on all machines are sequenced to provide a feasible schedule [1]. Over the years, various algorithms have been developed for the FJSP but general formula of FJSP can be represented based on the JSP. Consider a set of  $n$  jobs, represented as  $J = \{J_1, J_2, \dots, J_n\}$ , each job  $J_i$  ( $1 \leq i \leq n$ ) in  $J$  has a pre-defined number of operation (NOP)s, and must be operated on a selected machine from a machine set name  $M = \{M_1, M_2, \dots, M_M\}$ . There are two main differences between FJSP and JSP. First, there are  $n \times m$  operations when  $n$  jobs and  $m$  machines are available in the classical JSP. On the other hand, when  $n$  jobs and  $m$  machines are available, the NOPs may be larger or smaller than  $n \times m$  in FJSP. Second, an operation should be operated on a pre-defined machine in the classical JSP, while an operation may be operated by a set of machines in FJSP [2].

Although numerous studies related with the FJSPs are taken into account in literature, studies that consider flowtime as a performance metric for flexible job shops are scarce. Nasr and Elsayed [3] proposed two algorithms to minimize the mean flowtime in job shop scheduling while providing alternative machine routings. Tunalı [4] presented the impact of the selected routings in the system. The determination of the most appropriate routing for each part is considered as a critical issue in most of the studies. Chan et al. [5] used a fuzzy approach and investigated the influence of real-time multi-objective operation or routing selection on the performance of a hypothetical flexible manufacturing system (FMS). Hussain and Joshi [6] presented a two pass GA for a JSP with alternative routings. First, the alternatives are picked using GA. Then, the order and start time of jobs on the selected

alternatives are provided by means of a non-linear program. Moon et al. [7] developed a mixed integer linear programming and compared with the GA for the JSP with alternative routings. Saygin and Kilic [8] presented a real-time routing of parts in random FMSs that have machine, operation, and routing flexibility. In the study, a dissimilarity maximization method was implemented. Ozmutlu and Harmonosky [9] proposed a threshold based alternate routing to minimize mean flowtime in FMSs with routing flexibility and machines subject to breakdowns. Rossi and Dini [10] utilized an ant colony optimization based method to analyse the FJSP. Then, the effectiveness of proposed method was compared with other alternative methods using various benchmark instances. Buyurgan and Saygin [11] used analytical hierarchy process for determining dynamic performance measures to be used for part routing (PR) decisions and real-time scheduling. Shalaby et al. [12] developed a new routing rule for dynamic FJSP that includes the sequence-dependent setup times. Then, proposed methodology was compared with traditional routing rules using simulation experiments. In a network based manufacturing system, Manupati et al. [13] presented a scheme for generating optimal process plan (PP)s for multi jobs to minimize the makespan. Doh et al. [14] presented the FJSP with multiple PPs in which each job can be processed through alternative operations. In these plans, routing and operation flexibilities can be considered at the same time. Geyik and Dosdogru [15] used OvS method in order to solve the FJSP with random interarrivals and processing times. In order to generate optimal PPs, Manupati et al. [16] proposed an integration approach for networked manufacturing systems. Jiang et al. [17] presented a remanufacturing PP selection using fuzzy linear regression-based method and quality function deployment. Dosdogru et al. [18] integrated the GA and Monte Carlo method to analyse the stochastic flexible job shop scheduling. Chaudhry and Usman [19] presented the integrated process planning and scheduling. In the study, GA was used to select a process plan from a set of plans and to schedule the jobs on available resources.

Analyses of previous studies showed that OvS is a challenging alternative method to cope with noisy outputs and complex systems especially in the DSFJS. OvS algorithms that are found in the research literature try to move in relatively improving directions while utilizing some form of randomization to escape from local optimal solutions [20]. OvS algorithm is suitable when the objective function needs to be evaluated through simulation, and helps to find an optimal or a good solution. OvS algorithm generates solutions from the search space, estimates the performance measures of each sampled solution, moves to a promising solution or region and repeats these steps until some stopping criteria are satisfied [21]. Robust heuristics are one of the most common algorithms found in the integrated OvS tools. GA can be considered as one prominent example, but others such as tabu search, harmony search are available [22]. An extended analysis of literature related with metaheuristics for OvS was presented by Ólafsson [23].

DSFJS is one of the most important topics for managers and researchers. By considering current state of the companies, uncertainty and dynamic nature of the system should be considered from different perspectives. Although various methods are available in literature, finding the optimal solution is analytically solvable only under certain assumptions. In addition, DSFJS problems are generally not solvable in a reasonable amount of time with exact methods. In this case, OvS plays a key role due to the adaptability to proposed system. Furthermore, OvS has ability of capturing the advantages of both simulation and optimization based models simultaneously. In this paper, we proposed a new adaptable OvS to overcome the challenges of DSFJS problem under uncertainty. OvS is used to optimize PP and PR in DSFJS while minimizing the “total of average flowtimes of each part (TAFEP)” as the objective function. First, the best PP is selected by OvS for each part. Then, the best machine-operation (M-O) pairings is determined by proposed method. In this respect, GA is used to define PPs of each part and then select suitable machines for each operation of each part,

respectively. Then, simulation evaluates the performance of the determined PR by GA in terms of the considered performance measure. The rest of the paper is created as follows: Section 2 gives a detailed problem description. Simulation model assumptions and GA steps are also given in the section 2. Section 3 explains the experimental design that is constructed to investigate the effect of design parameters. In this section, computational results and analysis are commented. And, final part of the study gives the conclusion.

## 2. PROBLEM DESCRIPTION FOR OVS

Both PP and PR optimization problem in the DSFJS are handled to investigate the effectiveness of the OvS methodology. Suppose that there are several part types in a production system. There are several alternative PPs for each part and several alternative machines for each operation of parts with each PP considered. It should be noted that while one of the alternative PP include only one PP, the others include multi PPs as explained below. Here, PP selection is the problem of determining a PP for each part (i.e., operation sequences) and machine selection is the problem of selecting a machine for each operation among those associated with the operation. In main problem is to define the best PP and the best M-O pairing in order to minimize the “TAFEP” in the DSFJS. In this study, there are  $N$  parts  $P = \{p_1, p_2, \dots, p_N\}$  indexed by  $j$  and  $N$  operations  $\{O_{i1}, O_{i2}, \dots, O_{iN}\}$  indexed by  $i$ . Alternative PPs and/or processing sequences are pre-determined for each part.

Table I: Part PPs and processing sequence.

Parts		Number of operation : 3			Parts		Number of operation : 5					Parts		Number of operation : 8								
		Plans	Operation No				Plans	Operation No						Plans	Operation No							
Alternative Process Plans for Parts	Part 1	Plan1	1	2	3	Part 1	Plan1	1	2	4	5	3	Part 1	Plan1	2	1	3	5	4	7	8	6
		Plan2	2	1	3		Plan2	2	3	1	5	4		Plan2	2	1	4	3	6	5	7	8
		Plan3	3	2	1		Plan3	1	3	2	5	4		Plan3	7	4	8	1	3	2	5	6
	Part 2	Plan1	2	1	3	Part 2	Plan1	1	2	4	3	5	Part 2	Plan1	1	3	2	5	4	6	8	7
		Plan2	1	2	3		Plan2	2	3	1	5	4		Plan2	2	1	4	5	3	7	8	6
		Plan3	3	2	1		Plan3	2	4	5	3	1		Plan3	2	1	3	5	4	6	7	8
	Part 3	Plan1	1	2	3	Part 3	Plan1	4	3	1	2	5	Part 3	Plan1	1	3	4	2	6	5	7	8
		Plan2	3	2	1		Plan2	1	2	4	5	3		Plan2	6	8	7	1	3	4	2	5
		Plan3	2	1	3		Plan3	1	3	2	4	5		Plan3	1	2	4	3	6	5	7	8
	Part 4	Plan1	3	1	2	Part 4	Plan1	2	3	1	4	5	Part 4	Plan1	2	1	4	5	3	7	6	8
		Plan2	2	3	1		Plan2	1	2	4	5	3		Plan2	2	3	4	5	1	6	8	7
		Plan3	1	3	2		Plan3	2	1	4	3	5		Plan3	6	8	2	3	1	5	4	7
	Part 5	Plan1	1	2	3	Part 5	Plan1	2	1	4	5	3	Part 5	Plan1	1	3	2	4	5	7	8	6
		Plan2	3	2	1		Plan2	2	3	1	5	4		Plan2	3	6	5	8	7	1	2	4
		Plan3	2	3	1		Plan3	1	2	4	5	3		Plan3	2	1	4	3	6	7	5	8
	Part 6	Plan1	1	2	3	Part 6	Plan1	2	5	4	1	3	Part 6	Plan1	2	3	1	5	4	6	8	7
		Plan2	1	2	3		Plan2	2	3	1	5	4		Plan2	3	5	6	1	2	4	8	7
		Plan3	1	3	2		Plan3	1	2	4	3	5		Plan3	1	2	4	3	6	7	5	8
	Part 7	Plan1	2	1	3	Part 7	Plan1	1	3	2	5	4	Part 7	Plan1	6	5	8	7	2	3	1	4
		Plan2	2	3	1		Plan2	2	3	1	5	4		Plan2	1	3	2	5	4	6	8	7
		Plan3	3	1	2		Plan3	2	4	5	1	3		Plan3	4	5	7	1	3	2	6	8

Table I gives alternative PPs and/or processing sequences for all parts with 3, 5, and 8 numbers of operations. It should be noted that alternative PP table gives all possible plans for all parts and  $NOPs$  considered. This means that Plan 1 is available when the number of alternative process plan ( $NAPP$ ) is 1. Similarly, when the number of alternative plan is 2, there are two alternative processing sequences including Plan 2’s processing sequence as the second alternative, and Plan 3 is available as the third alternative when the  $NAPP$  is 3. For example, consider Part 1 with 3 numbers of operations from Table I. Here, if the  $NAPP$  is 1,

Part 1 has only one processing sequence, i.e., 1, 2, 3 which means  $\{O_{11}, O_{12}, O_{13}\}$  (Plan 1). If the *NAPP* is 2, Part 1 have two alternative processing sequences, i.e., 1, 2, 3 (Plan 1) and 2, 1, 3 (Plan 2). Similarly, if the *NAPP* is 3, Part 1 have three alternative processing sequences, i.e., 1, 2, 3 (Plan 1), 2, 1, 3 (Plan 2) and 3, 2, 1 (Plan 3).

Table II: Operation processing time of each operation.

Parts	Machines	*Op. 1	Op. 2	Op. 3	Op. 4	Op. 5	Op. 6	Op. 7	Op. 8	
Operation Processing Times of Parts at each Alternative Machine	Part 1	M1	**(2,4,6)	(2,4,6)	(1,3,5)	(1,3,5)	(5,7,9)	(4,6,8)	(4,6,8)	(4,6,8)
		M2	(4,6,8)	(5,7,9)	(5,7,9)	(1,3,5)	(5,7,9)	(2,4,6)	(4,6,8)	(4,6,8)
		M3	(1,3,5)	(1,3,5)	(3,5,7)	(3,5,7)	(5,7,9)	(4,6,8)	(1,3,5)	(4,6,8)
		M4	(1,3,5)	(2,4,6)	(5,7,9)	(1,3,5)	(3,5,7)	(4,6,8)	(1,3,5)	(5,7,9)
		M5	(2,4,6)	(2,4,6)	(2,4,6)	(1,3,5)	(1,3,5)	(4,6,8)	(4,6,8)	(5,7,9)
	Part 2	M1	(2,4,6)	(2,4,6)	(1,3,5)	(3,5,7)	(1,3,5)	(2,4,6)	(1,3,5)	(3,5,7)
		M2	(1,3,5)	(1,3,5)	(4,6,8)	(3,5,7)	(2,4,6)	(2,4,6)	(2,4,6)	(2,4,6)
		M3	(2,4,6)	(2,4,6)	(2,4,6)	(3,5,7)	(4,6,8)	(4,6,8)	(1,3,5)	(3,5,7)
		M4	(1,3,5)	(3,5,7)	(2,4,6)	(2,4,6)	(1,3,5)	(3,5,7)	(3,5,7)	(4,6,8)
		M5	(2,4,6)	(1,3,5)	(4,6,8)	(3,5,7)	(2,4,6)	(2,4,6)	(1,3,5)	(5,7,9)
	Part 3	M1	(4,6,8)	(4,6,8)	(5,7,9)	(5,7,9)	(1,3,5)	(4,6,8)	(4,6,8)	(3,5,7)
		M2	(3,5,7)	(2,4,6)	(3,5,7)	(5,7,9)	(2,4,6)	(3,5,7)	(4,6,8)	(5,7,9)
		M3	(4,6,8)	(4,6,8)	(3,5,7)	(4,6,8)	(1,3,5)	(5,7,9)	(4,6,8)	(3,5,7)
		M4	(4,6,8)	(4,6,8)	(5,7,9)	(4,6,8)	(1,3,5)	(3,5,7)	(3,5,7)	(1,3,5)
		M5	(4,6,8)	(2,4,6)	(3,5,7)	(5,7,9)	(1,3,5)	(5,7,9)	(1,3,5)	(5,7,9)
	Part 4	M1	(3,5,7)	(1,3,5)	(2,4,6)	(2,4,6)	(1,3,5)	(3,5,7)	(5,7,9)	(1,3,5)
		M2	(4,6,8)	(2,4,6)	(4,6,8)	(2,4,6)	(4,6,8)	(4,6,8)	(3,5,7)	(1,3,5)
		M3	(4,6,8)	(4,6,8)	(1,3,5)	(2,4,6)	(1,3,5)	(4,6,8)	(4,6,8)	(3,5,7)
		M4	(4,6,8)	(2,4,6)	(2,4,6)	(5,7,9)	(5,7,9)	(3,5,7)	(4,6,8)	(1,3,5)
		M5	(4,6,8)	(1,3,5)	(2,4,6)	(4,6,8)	(5,7,9)	(4,6,8)	(5,7,9)	(3,5,7)
	Part 5	M1	(3,5,7)	(4,6,8)	(1,3,5)	(3,5,7)	(1,3,5)	(3,5,7)	(2,4,6)	(2,4,6)
		M2	(3,5,7)	(5,7,9)	(3,5,7)	(3,5,7)	(1,3,5)	(5,7,9)	(2,4,6)	(4,6,8)
		M3	(1,3,5)	(5,7,9)	(1,3,5)	(3,5,7)	(5,7,9)	(3,5,7)	(2,4,6)	(3,5,7)
		M4	(3,5,7)	(5,7,9)	(1,3,5)	(3,5,7)	(1,3,5)	(1,3,5)	(3,5,7)	(2,4,6)
		M5	(1,3,5)	(5,7,9)	(3,5,7)	(5,7,9)	(5,7,9)	(3,5,7)	(3,5,7)	(4,6,8)
	Part 6	M1	(2,4,6)	(4,6,8)	(4,6,8)	(3,5,7)	(5,7,9)	(1,3,5)	(3,5,7)	(2,4,6)
		M2	(3,5,7)	(3,5,7)	(4,6,8)	(3,5,7)	(3,5,7)	(1,3,5)	(4,6,8)	(2,4,6)
		M3	(3,5,7)	(2,4,6)	(2,4,6)	(2,4,6)	(1,3,5)	(1,3,5)	(5,7,9)	(2,4,6)
		M4	(1,3,5)	(4,6,8)	(2,4,6)	(4,6,8)	(5,7,9)	(3,5,7)	(5,7,9)	(5,7,9)
		M5	(3,5,7)	(3,5,7)	(2,4,6)	(3,5,7)	(1,3,5)	(1,3,5)	(3,5,7)	(2,4,6)
	Part 7	M1	(1,3,5)	(2,4,6)	(4,6,8)	(4,6,8)	(4,6,8)	(2,4,6)	(1,3,5)	(3,5,7)
		M2	(1,3,5)	(1,3,5)	(5,7,9)	(4,6,8)	(2,4,6)	(1,3,5)	(3,5,7)	(3,5,7)
		M3	(1,3,5)	(4,6,8)	(3,5,7)	(4,6,8)	(1,3,5)	(1,3,5)	(2,4,6)	(3,5,7)
		M4	(1,3,5)	(2,4,6)	(4,6,8)	(4,6,8)	(2,4,6)	(2,4,6)	(2,4,6)	(5,7,9)
		M5	(1,3,5)	(4,6,8)	(3,5,7)	(4,6,8)	(2,4,6)	(1,3,5)	(2,4,6)	(3,5,7)

Remark: \*Op. represents operation, \*\* the parameters of triangular distribution.

Table II gives the alternative machine sets for each operation. Three levels for flexibility are adjusted according to the number of alternative machines for each operation. Operation processing times is considered as triangular distribution (Table II). The problem is considered as two separate phases. In the first phase, PPs is determined for each part and then appropriate machine is selected for each operation of each part, respectively. This phase is called the optimization phase. Second is the simulation phase where new sets of values for decision variables produced by the first phase are used. It should be noted that while the sets of values in the second phase are evaluating, the system dynamics are to be taken into account. In this respect, GA is created to identify new values for the selected decision variables, and a

simulation model is constructed to evaluate these new values of decision variables. This iterative process successively produces new sets of values for the decision variables. This process continues until some termination criteria are met – generally it is determined as a maximum number of simulations or GA terminates when there is no improvement on the objective function value.

## 2.1 Simulation phase

In simulation, the system performance can be observed under different sets of parameters. In addition, the parameters of a simulation model can be easily changed. Hence, it is natural to try to determine the set of parameters that gives the optimal results for the considered system. In this paper, simulation is used to analyse the performance of the PRs (i.e., machine & operation pairings) obtained by optimization module (GA). A computer model of DSFJS is coded in SIMIO 6.105.11267 Enterprise Edition. As stated in the study of Fu [24], "decision variables" are taken as inputs and "objective function" denotes the outputs. In this case, decision variables are alternative PP for each part and the alternative machines for each operation of each part. Objective function is the "TAFEP". Note that alternative machines for each operation of any part depend on the considered PP.

Simulation model properties are determined as follows:

- Parts dynamically arrive to the shop and the arrival rate is exponentially distributed with a mean of 1.5 hours.
- At each arrival only one part arrives at the system (i.e., no batches considered).
- Jobs are released to the shop as soon as they arrive to the system.
- FIFO dispatching rule is considered in front of all machine queues.
- Preemption is not allowed.
- No load balancing is taken into account.
- A machine cannot perform more than one operation at a time.
- Consecutive operations of parts can be processed on the same machine.
- The setup times and transportation times are not taken into account.
- Machine breakdowns are not considered.
- Simulation models runs until completion of a total of 18000 units for each part.

## 2.2 Optimization phase

Due to its prevalence and effectiveness in simulation, GAs will continue to be an important part of OvS for solving difficult problems [23]. GA, coded in Microsoft Visual C# Express 2012, is integrated with the model and changes the values of the decision variables of the model directly (i.e., by this way GA generates candidate solutions). Simulation model evaluates these new values of decision variables. The results of the simulation run are returned back to GA. Then, GA once more tries to determine better decision variables to improve the performance of the model. This iterative process stops when the objective value of GA has stopped improving. In subsequent sections, GA is explained, and the steps of the algorithm are given.

## 2.3 Genetic algorithm

In this paper, GA is created to determine the appropriate PP for each part and also to specify the most appropriate M-O pairs among the machine sets for each operation of each part, respectively. To solve the PP selection problem and M-O matching by using the technique of GA, the details of the GA should be designed according to the nature of the problem. This includes the representation of the chromosomes, fitness evaluation, initial population, selection, crossover and mutation.

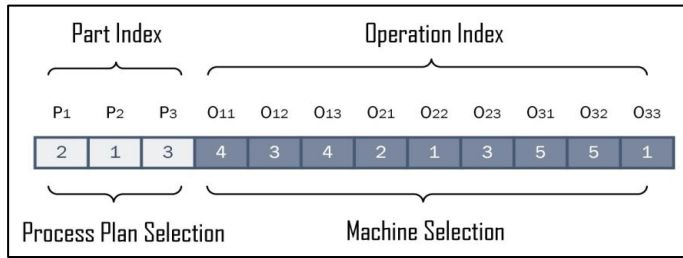


Figure 1: The chromosome representation.

In GA, the chromosome structure is considered as a two parts: (1) PP selection (i.e., operation sequence) and (2) M-O matching (i.e., PR). The selection of PP for each part is represented in the first part of the chromosome whose length is equal to the number of part ( $NP$ )s. The second part denotes the selected machines for each operation of each part. Length of the second part of the chromosome is equal to the total  $NOPs$ . Fig. 1 shows an illustration of a chromosome representation of a 3 parts and 3 operations problems.

From literature, it is apparently seen that FJSP has generally been focused on minimizing makespan. The studies related to flowtime minimization for FJSP are scarce. For each job, flowtime can be calculated by using Eq. (1).

$$F_i = \sum_{j \in S(i)} s_{ij} + m_{ij} + w_{ij} + p_{ij} \quad (1)$$

where  $i$  is job type and  $j$  is station no.;  $s_{ij}$  is setup time needed by job  $i$  at station  $j$ ;  $m_{ij}$  is transportation time necessary for moving job  $i$  from station  $j$  to the next station on its route;  $w_{ij}$  is queue waiting time of job  $i$  at station  $j$ 's queue;  $p_{ij}$  is processing time of job  $i$  at station  $j$ ;  $S(i)$  is the set of stations which are placed on job  $i$ 's route [25]. From simulation runs we get an estimate of flowtime of each part, shown as  $\hat{F}_i$ . Then, our objective function to be minimized called "TAFEP" can be expressed by Eq. (2) as follows:

$$\sum_{i=1}^{\# \text{ of Parts}} \hat{F}_i \quad (2)$$

The fitness evaluation operation of GA calculates the fitness value of each individual considering the objective function (i.e., TAFEP). In proposed GA, initial population generation consists of two steps. First, PPs are selected for each part randomly. So, processing sequences of all parts are to be determined. Then, machines are assigned to each operation appropriately by considering the part's operations' alternative machines. When the number of initial solutions is equal to the population size, the initial population is fed into the simulation model to be evaluated.

Then, individuals are selected for reproduction and placed in the mating pool. In this study, individuals are selected by using roulette wheel selection. Roulette wheel selects individuals according to its fitness function. Fitter individuals are more likely to be chosen. In each generation, best individual is reserved and translated to the next generation. Finally, two-point crossover and mutation are applied to obtain new high quality solutions. In this paper, the value of number of iterations is 50, the population size is 20, and the mutation probability value is 0.05.

### **3. EXPERIMENTAL DESIGN**

The design has three levels for four main factors considered. Noted that both level of flexibility 2 and level of flexibility 3 incorporate all previous levels of flexibilities' alternative

machines for each operation. This means that with level of flexibility 1 each operation has no alternative machine(s), with level of flexibility 2 each operation have three alternative machines including level of flexibility 1's machines, and with level of flexibility 3 each operation have five alternative machines including level of flexibility 2's machines. The natural and coded levels of each considered main factor summarized in Table III.

Table III: Levels of experimental factors.

Factors	Levels	Range of variables					
		Low Level		Intermediate Level		High Level	
		Natural	Coded	Natural	Coded	Natural	Coded
Flexibility	3	1	1	3	2	5	3
Number of parts	3	3	1	5	2	7	3
Number of operations	3	3	1	5	2	8	3
Number of process plans	3	Plan1	1	Plan 2	2	Plan 3	3

As a result, 81 experiments are necessary (i.e.,  $3 \times 3 \times 3 = 81$ ) to analyse all factor level combinations. The simulation experiments are performed using 10 replications of each treatment in order to minimize the variability in the results. Common random numbers are also utilized to obtain the same experimental condition across the runs for each factor combination. Note that,  $\alpha = 0.05$  is used in evaluating statistical significance. Only several results of experimental design are given in Table IV since 81 experiments take up too much space.

Table IV: Several results of experimental design.

Experiment No.	Number of parts	Number of operations	Flexibility	Number of process plans	Result
1	3	3	1	1	217,0503
2	3	3	1	2	201,9828
3	3	3	1	3	188,7902
4	3	5	1	1	270,4483
5	3	5	1	2	236,9860

### 3.1 Computational results and analysis

Analysis of variance (ANOVA) or  $F$  test is a significance test. The ANOVA procedure, when its assumptions are satisfied, utilizes all available information. ANOVA has three assumptions. First, the observations are assumed as independent. Second, the variances of the groups are equal. Final assumption is that the dependent variable is normally distributed for each group [26]. When there are equal numbers of subjects in each experimental condition, ANOVA is fairly robust with respect to certain assumption violations [27]. In addition, some parametric statistics, such as the two tailed  $t$  test and ANOVA, are quite robust so even a skewness of more than  $\pm 1$  may not change the results much [26]. In this study, data is firstly analysed for normality by using Q-Q plot and skewness is determined as 1.131. The reason for this is that TAFEP actually comes from more than one process. It is not normally distributed and they look multimodal which are overlapped. One of the solutions is to determine which average flowtime value triggers the problem and stratify the data. However, TAFEP composed of too many value. Data cannot be checked again for normality after stratified processes. Hence, stratified processes cannot be applied in the study. Nevertheless, TAFEP should be transformed to normal distribution with constant variance before ANOVA is used. For these purposes, data transformations are commonly-used tools. In literature, various data transformations type is available but Osborne [28] showed that Box-Cox can be a

potential best practice where normalizing data or equalizing variance is desired. Moreover, Box-Cox is highly recommended in Weibull Data and the value of TAFEP resembles Weibull Data. Hence, Box-Cox transformation can be used to get the normality in this study. Box-Cox transformation is applied by using XLSTAT and automatically determined best transformation parameter is 0.013. Transformed data is then analysed for normality. In Shapiro-Wilk,  $p$  value (0.517) is greater than the significance level  $\alpha = 0.05$ , one cannot reject the null hypothesis  $H_0$ . This means that the variable from which the sample is extracted follows a Normal distribution. After normality, variance is analysed with Levene test and result of the test shows that the assumption of homogeneity of variances has been violated. Fortunately, ANOVA is calculated by means of SPSS that uses the regression approach. Therefore, this problem is not important. Consequently, ANOVA is applied and detailed information about the system is analysed.

Effects of the four main factors are given in Fig. 2. It is seen that the system performances are affected by all main factors. It is also important to note that some of the factor interactions do not have remarkable effect on the TAFEP and a detailed analysis of these insignificant interaction effects are explained by interaction plots in the subsequent sections. The plots for means of the “TAFEP” are shown Fig. 3 (a, b, c, d, e, and f). To assist in analysing the interaction effect of considered factors, it is helpful to construct a graph of the average responses at each level of all factor combinations. Note that, for each interaction effect graphs given only interaction effect of considered two factors will be analysed. Thus, other factors’ effects will be averaged for each level of the considered factors.

<b>Tests of Between-Subjects Effects</b>					
Dependent Variable: Total of average flowtimes of each part					
Source	Type III Sum of Squares	df (degree of freedom)	Mean Square	F	Sig.
Corrected Model	24,487 <sup>a</sup>	64	0.383	477.846	0
Intercept	2919.901	1	2919.901	3646718.111	0
# of alternative process plans	0.082	2	0.041	51.199	0
# of parts	12.249	2	6.125	7649.046	0
# of operations	8.889	2	4.444	5550.716	0
Flexibility level	2.996	2	1.498	1870.588	0

Figure 2: Tests of between-subjects effects.

In Fig. 3 a, interaction effect of flexibility and *NOPs* on “TAFEP” performance of the shop is shown. The significant interaction is indicated by the lack of parallelism of the lines. As the level of flexibility increases, “TAFEP” values with all levels of *NOPs* decreases. Nevertheless, as the level of flexibility increases from low to intermediate and intermediate to high, “TAFEP” values with low level of *NOPs* (i.e., where the *NOPs* is 3) significantly decreases in comparison with its other levels (i.e., where the numbers of operations are 5 and 8). For low level of flexibility, “TAFEP” performances of the low level of *NOPs* and the intermediate level of *NOPs* differs slightly. However, the magnitude of the difference between these two levels of *NOPs* increases for higher levels of flexibility which means that low level of *NOPs* benefits from increased level of flexibility much more than the other levels of the *NOPs*. Apparently, shorter “TAFEP” is attained at low level of *NOPs* regardless of the levels of flexibility. Also, high level of flexibility provides the best total of average flowtimes performance for each level of *NOPs* considered.



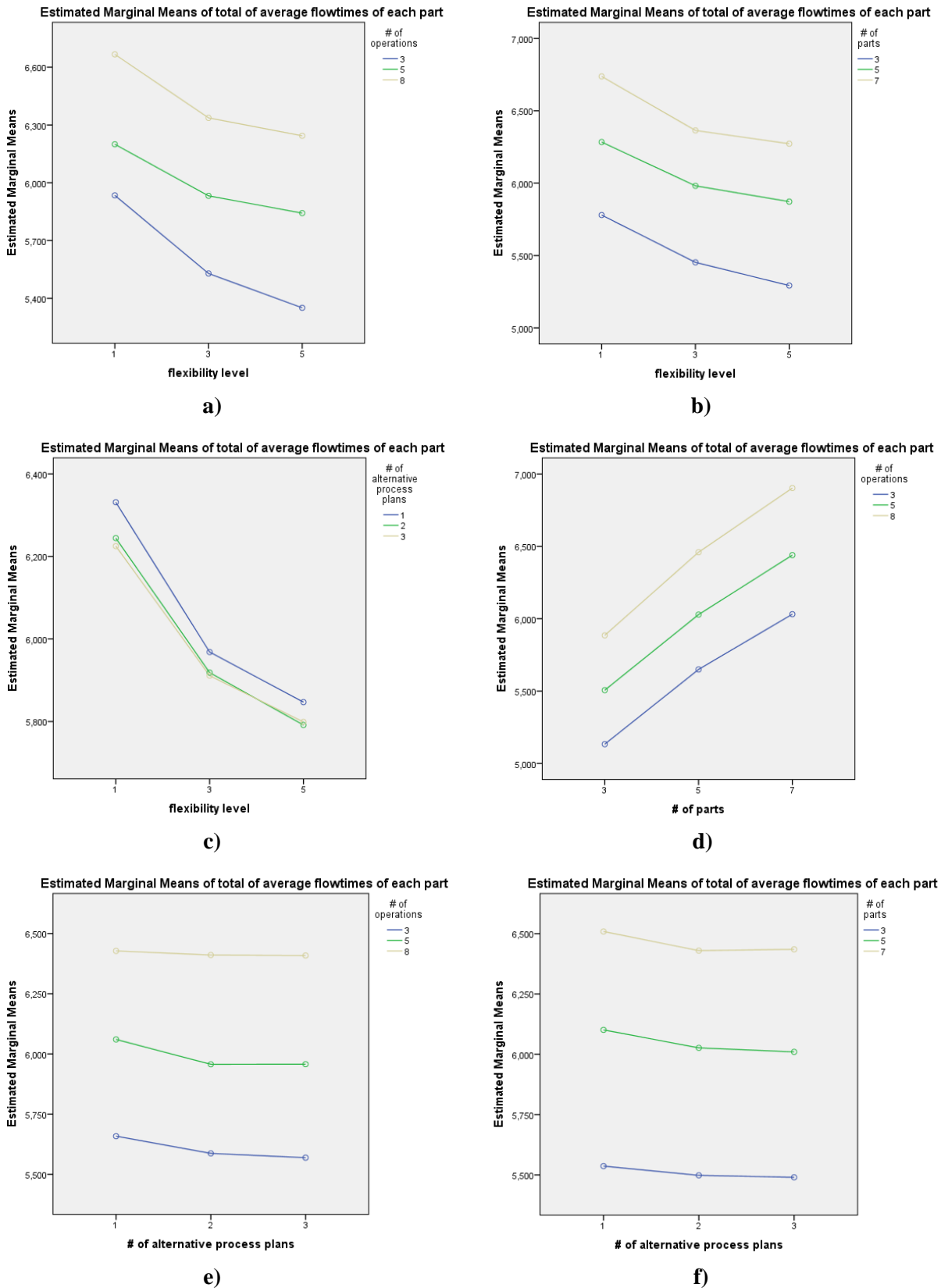


Figure 3: Interaction effects; a) *FL* – *NOPs*, b) *FL* – *NPs*, c) *FL* – *NAPPs*, d) *NPs* – *NOPs*, e) *NAPPs* – *NOPs*, f) *NAPPs* – *NPs*.

In Fig. 3 b, interaction effect of flexibility and the *NPs* on “TAFEP” performance of the shop is shown. In general, shorter “TAFEP” is attained at high level of flexibility regardless

of the levels of *NPs*, whereas higher “TAFEP” is attained at low level of flexibility. Changing from low to higher levels of flexibility, “TAFEP” with all levels of *NPs* actually decreases. Note that, the magnitude of the difference between the levels of the *NPs* remains almost the same for each flexibility level (*FL*). Again, it is clearly seen that low level of *NPs* exhibits the best “TAFEP” performance regardless of *FL*. Also, high level of flexibility provides the best total of average flowtimes performance for each level of *NPs* considered.

In Fig. 3 c, interaction effect of the flexibility and the *NAPPs* on “TAFEP” performance of the shop is shown. The shorter “TAFEP” is generally attained at high level of flexibility regardless of the *NAPPs*. Changing from low to higher levels of flexibility, “TAFEP” performance with all *NAPPs* actually improves. Note that, for each level of flexibility, “TAFEP” value for all *NAPPs* remains almost the same. Noted that as the level of flexibility increases the improvement on “TAFEP” decreases for each *NAPP* considered. This reveals that if there were, any higher levels of flexibility would get only marginal improvements on the “TAFEP” performance of the shop. *NAPPs* 2 and 3 exhibit almost the same “TAFEP” performance for all levels of flexibility whereas it is apparently seen that the PP 1’s “TAFEP” performance is a bit worse than the others for all considered levels of flexibility. Again, as the level of the flexibility increases the “TAFEP” performance of all *NAPPs* improves regardless of the alternative number of PPs. Thus, it is apparently seen that the higher *FL* improves the system performance for each level of the *NAPPs*.

In Fig. 3 d, interaction effect of *NPs* and *NOPs* on “TAFEP” performance of the shop is shown. In general, shorter “TAFEP” is attained at low level of *NPs* for each level of *NOPs* considered, whereas “TAFEP” performance deteriorates as the level of *NPs* increases for each level of *NOPs*. The magnitude of deterioration of “TAFEP” performance slightly decreases as the level of *NPs* increases for each level of *NOPs* considered. In addition, the lower levels of the *NPs* improve the system performance for each level of the *NOPs*. Consequently, the best “TAFEP” performance is attained at low level of *NOPs* for each level of the *NPs*.

In Fig. 3 e, interaction effect of *NAPPs* and *NOPs* on “TAFEP” performance of the shop is shown. From Fig. 3 e, as indicated by the parallelism of the lines, there is no significant interaction effect between *NAPPs* and *NOPs*. However, from the figure it can be said that there is a weak interaction particularly between *NAPPs* and low and intermediate level of *NOPs*. At low level of *NOPs*, as the level of *NAPPs* increases from low to intermediate “TAFEP” performance slightly improves. However, as the level of *NAPPs* increases from intermediate to high “TAFEP” performance of the shop only marginally improves. This is the case for intermediate level of *NOPs* but with a twist, as well. This time as the level of flexibility increases from intermediate to high the “TAFEP” performance of the shop remains essentially the same. Note that for high level of *NOPs* “TAFEP” performance of the shop barely improves as the level of *NAPPs* increases. Consequently, it can be said that there is negligible *NAPPs* and *NOPs* interaction effect on “TAFEP” performance. Nevertheless, the best performance is attained with low level of *NOPs* for all levels of *NAPPs*.

From Fig. 3 f, as indicated by the parallelism of the lines, there is no significant interaction effect between *NAPPs* and *NPs*. At low level of *NPs*, as the level of *NAPPs* increases from low to intermediate “TAFEP” performance slightly improves. However, as the level of *NAPPs* increases from intermediate to high “TAFEP” performance of the shop only marginally improves. This is the case for intermediate level of *NOPs*, as well. At high level of *NPs*, as the level of *NAPPs* increases from low to intermediate the “TAFEP” performance of the shop slightly increases. But, as the level of *NAPPs* increases from intermediate to high the performance of the shop slightly deteriorates. Consequently, it can be said that there is negligible *NAPPs* and *NPs* interaction effect on “TAFEP” performance. Nevertheless, the best performance is attained with low level of *NPs* for each level of *NAPPs*.

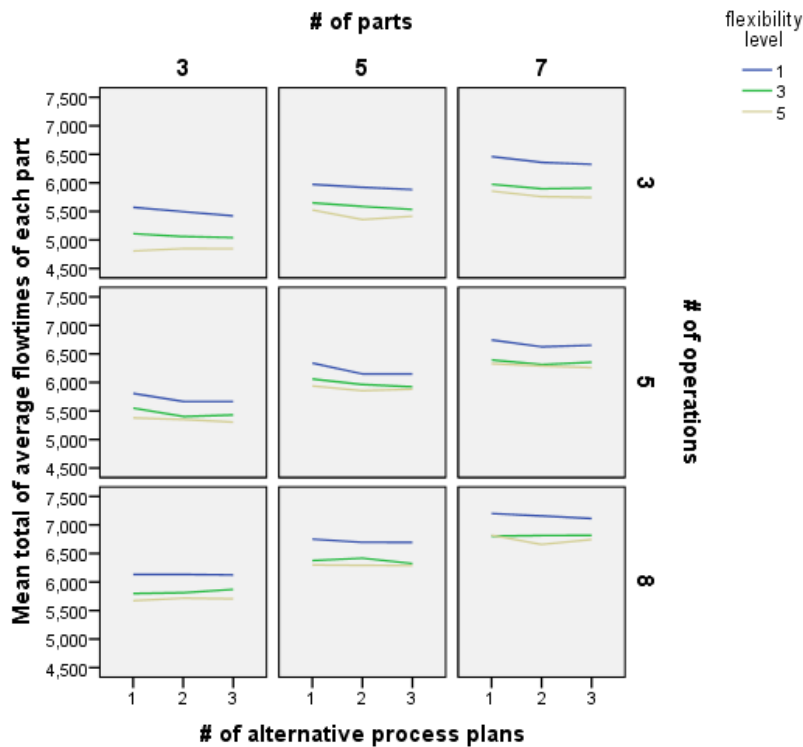


Figure 4: Interaction effect of *FL*, *NPs*, *NAPPs*, and *NOPs*.

Interaction effect of *FL*, *NPs*, *NAPPs*, and *NOPs* on system performance is given in Fig. 4. It is clearly seen that increasing *FL* improves the “TAFEP”. As the level of *NOPs* and *NPs* increases, the “TAFEP” performance of the shop deteriorates. Results revealed that “TAFEP” performance of the FJSP are greatly influenced by main factors and almost all factor interactions mentioned above.

#### **4. CONCLUSION**

In this study both PP and PR optimization problem in the DSFJS is handled using OvS methodology. In addition, experimental design is carried out to analyse the effects of some predetermined main factors and factors’ interactions on “total of average of flowtimes for each part” performance of the shop. The design includes *FL* of the shop, *NOPs*, *NPs*, and *NAPPs*. Thus, 81 (i.e.  $3 \times 3 \times 3 \times 3$ ) factor combinations are tested in the experiments. Then, ANOVA is used to analyse the performance of the system and evaluate the relationships between the considered factors. ANOVA demonstrated that system performance of the dynamic stochastic FJSP is greatly influenced by main factors and some factor interactions.

One of the main conclusions is that flexibility does positively affects the performance of the shop to a degree. But, it should be noted that higher levels of flexibility would get only marginal improvements on the “TAFEP” performance of the shop. The performance improvement with higher levels of flexibility is off set by the resultant complexity (i.e., computational time) of higher *FLs*. As a result, one has to fine tune the level of flexibility by considering the main factors and factor interactions for the problem at hand as a whole. In addition to flexibility, it is also determined that system performance is directly affected by other factors including number of parts, number of operations, and number of process plans. For example, when the levels of number of operations and number of parts increase, the “TAFEP” performance of the shop decreases.

Another significant conclusion is that using OvS methodology enables getting high quality solutions for dynamic stochastic FJSP that is known to be strongly NP hard. Solution

methodology of OvS requires much more time to develop and function evaluations are computationally expensive but, in general there is no another option for practical settings. Many real life problems in optimization are too complex to be given tractable mathematical formulations. This difficulty gives rise to a remarkable growth in utilization of OvS methodology as a solution tool not only in academic world but also in practical settings [29-31]. The results reveal that OvS methodology is a promising candidate in the solution of challenging real life problems and expected to be well suited to a wide range of application areas.

## **REFERENCES**

- [1] Wang, L.; Zhou, G.; Xu, Y.; Liu, M. (2013). A hybrid artificial bee colony algorithm for the fuzzy flexible job-shop scheduling problem, *International Journal of Production Research*, Vol. 51, No. 12, 3593-3608, doi:[10.1080/00207543.2012.754549](https://doi.org/10.1080/00207543.2012.754549)
- [2] Li, J.-Q.; Pan, Q.-K.; Xie, S.-X. (2010). A hybrid variable neighborhood search algorithm for solving multi-objective flexible job shop problems, *Computer Science and Information Systems*, Vol. 7, No. 4, 907-930, doi:[10.2298/CSIS090608017L](https://doi.org/10.2298/CSIS090608017L)
- [3] Nasr, N.; Elsayed, E. A. (1990). Job shop scheduling with alternative machines, *International Journal of Production Research*, Vol. 28, No. 9, 1595-1609, doi:[10.1080/00207549008942818](https://doi.org/10.1080/00207549008942818)
- [4] Tunali, S. (1997). Evaluation of alternate routing policies in scheduling a job-shop type FMS, *Computers & Industrial engineering*, Vol. 32, No. 2, 243-250, doi:[10.1016/s0360-8352\(96\)00306-3](https://doi.org/10.1016/s0360-8352(96)00306-3)
- [5] Chan, F. T. S.; Kazerooni, A.; Abhary, K. (1997). A fuzzy approach to operation selection, *Engineering Applications of Artificial Intelligence*, Vol. 10, No. 4, 345-356, doi:[10.1016/s0952-1976\(97\)80003-6](https://doi.org/10.1016/s0952-1976(97)80003-6)
- [6] Hussain, M. F.; Joshi, S. B. (1998). A genetic algorithm for job shop scheduling problems with alternate routing, *Proceedings of the 1998 International Conference on Systems, Man, and Cybernetics*, 2225-2230, doi:[10.1109/ICSMC.1998.724986](https://doi.org/10.1109/ICSMC.1998.724986)
- [7] Moon, I.; Lee, S.; Bae, H. (2008). Genetic algorithms for job shop scheduling problems with alternative routings, *International Journal of Production Research*, Vol. 46, No. 10, 2695-2705, doi:[10.1080/00207540701244820](https://doi.org/10.1080/00207540701244820)
- [8] Saygin, C.; Kilic, S. (2004). Dissimilarity maximization method for real-time routing of parts in random flexible manufacturing systems, *International Journal of Flexible Manufacturing Systems*, Vol. 16, No. 2, 169-182, doi:[10.1023/b:flex.0000044839.27404.92](https://doi.org/10.1023/b:flex.0000044839.27404.92)
- [9] Ozmutlu, S.; Harmonosky, C. M. (2005). A real-time methodology for minimizing mean flowtime in FMSs with routing flexibility: Threshold-based alternate routing, *European Journal of Operational Research*, Vol. 166, No. 2, 369-384, doi:[10.1016/j.ejor.2004.01.044](https://doi.org/10.1016/j.ejor.2004.01.044)
- [10] Rossi, A.; Dini, G. (2001). An evolutionary approach to complex job-shop and flexible manufacturing system scheduling, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. 215, No. 2, 233-245, doi:[10.1243/0954405011515181](https://doi.org/10.1243/0954405011515181)
- [11] Buyurgan, N.; Saygin, C. (2008). Application of the analytical hierarchy process for real-time scheduling and part routing in advanced manufacturing systems, *Journal of Manufacturing Systems*, Vol. 27, No. 3, 101-110, doi:[10.1016/j.jmsy.2008.08.002](https://doi.org/10.1016/j.jmsy.2008.08.002)
- [12] Shalaby, M. A.; Abdelmaguid, T. F.; Abdelrasol, Z. Y. (2012). New routing rules for dynamic flexible job shop scheduling with sequence-dependent setup times, *Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management*, 747-756
- [13] Manupati, V. K.; Deo, S.; Cheikhrouhou, N.; Tiwari, M. K. (2012). Optimal process plan selection in networked based manufacturing using game-theoretic approach, *International Journal of Production Research*, Vol. 50, No. 18, 5239-5258, doi:[10.1080/00207543.2012.682181](https://doi.org/10.1080/00207543.2012.682181)
- [14] Doh, H.-H.; Yu, J.-M.; Kim, J.-S.; Lee, D.-H.; Nam, S.-H. (2013). A priority scheduling approach for flexible job shops with multiple process plans, *International Journal of Production Research*, Vol. 51, No. 12, 3748-3764, doi:[10.1080/00207543.2013.765074](https://doi.org/10.1080/00207543.2013.765074)

- [15] Geyik, F.; Dosdoğru, A. T. (2013). Process plan and part routing optimization in a dynamic flexible job shop scheduling environment: an optimization via simulation approach, *Neural Computing and Applications*, Vol. 23, No. 6, 1631-1641, doi:[10.1007/s00521-012-1119-7](https://doi.org/10.1007/s00521-012-1119-7)
- [16] Manupati, V. K.; Thakkar, J. J.; Wong, K. Y.; Tiwari, M. K. (2013). Near optimal process plan selection for multiple jobs in networked based manufacturing using multi-objective evolutionary algorithms, *Computers & Industrial Engineering*, Vol. 66, No. 1, 63-76, doi:[10.1016/j.cie.2013.06.004](https://doi.org/10.1016/j.cie.2013.06.004)
- [17] Jiang, Z.; Fan, Z.; Sutherland, J. W.; Zhang, H.; Zhang, X. (2014). Development of an optimal method for remanufacturing process plan selection, *International Journal of Advanced Manufacturing Technology*, Vol. 72, No. 9-12, 1551-1558, doi:[10.1007/s00170-014-5783-x](https://doi.org/10.1007/s00170-014-5783-x)
- [18] Dosdoğru, A. T.; Göçken, M.; Geyik, F. (2015). Integration of genetic algorithm and Monte Carlo to analyze the effect of routing flexibility, *International Journal of Advanced Manufacturing Technology*, Vol. 81, No. 5-8, 1379-1389, doi:[10.1007/s00170-015-7247-3](https://doi.org/10.1007/s00170-015-7247-3)
- [19] Chaudhry, I. A.; Usman, M. (2017). Integrated process planning and scheduling using genetic algorithms, *Technical Gazette*, Vol. 24, No. 5, 1401-1409, doi:[10.17559/TV-20151121212910](https://doi.org/10.17559/TV-20151121212910)
- [20] Pichitlamken, J.; Nelson, B. L. (2003). A combined procedure for optimization via simulation, *ACM Transactions on Modeling and Computer Simulation*, Vol. 13, No. 2, 155-179, doi:[10.1145/858481.858485](https://doi.org/10.1145/858481.858485)
- [21] Park, C.; Kim, S.-H. (2011). Handling stochastic constraints in discrete optimization via simulation, *Proceedings of the 2011 Winter Simulation Conference*, 4212-4221, doi:[10.1109/WSC.2011.6148109](https://doi.org/10.1109/WSC.2011.6148109)
- [22] Nelson, B. L. (2010). Optimization via simulation over discrete decision variables, Hasenbein, J. J. (Ed.), *Risk and Optimization in an Uncertain World*, Tutorials in Operations Research, INFORMS, Catonsville, 193-207, doi:[10.1287/educ.1100.0069](https://doi.org/10.1287/educ.1100.0069)
- [23] Ólafsson, S. (2006). Metaheuristics, Henderson, S. G.; Nelson, B. L. (Eds.), *Handbooks in Operations Research and Management Science: Simulation*, Vol. 13, Elsevier, Amsterdam, 633-654, doi:[10.1016/S0927-0507\(06\)13021-2](https://doi.org/10.1016/S0927-0507(06)13021-2)
- [24] Fu, M. C. (2002). Optimization for simulation: Theory vs. practice, *INFORMS Journal on Computing*, Vol. 14, No. 3, 192-215, doi:[10.1287/ijoc.14.3.192.113](https://doi.org/10.1287/ijoc.14.3.192.113)
- [25] Baykasoğlu, A.; Göçken, M.; Unutmaz, Z. (2008). New approaches to due date assignment in job shops, *European Journal of Operational Research*, Vol. 187, No. 1, 31-45, doi:[10.1016/j.ejor.2007.02.020](https://doi.org/10.1016/j.ejor.2007.02.020)
- [26] Leech, N. L.; Barrett, K. C.; Morgan, G. A. (2005). *SPSS for Intermediate Statistics: Use and Interpretation*, 2<sup>nd</sup> edition, Lawrence Erlbaum, Mahwah
- [27] Khan, A.; Rayner, G. D. (2003). Robustness to non-normality of common tests for the many-sample location problem, *Journal of Applied Mathematics and Decision Sciences*, Vol. 7, No. 4, 187-206, doi:[10.1155/S1173912603000178](https://doi.org/10.1155/S1173912603000178)
- [28] Osborne, J. W. (2010). Improving your data transformations: Applying the Box-Cox transformation, *Practical Assessment, Research & Evaluation*, Vol. 15, Paper 12, 9 pages
- [29] Gocken, M.; Dosdogru, A. T.; Boru, A. (2017). Optimization via simulation for inventory control policies and supplier selection, *International Journal of Simulation Modelling*, Vol. 16, No. 2, 241-252, doi:[10.2507/IJSIMM16\(2\)5.375](https://doi.org/10.2507/IJSIMM16(2)5.375)
- [30] Meng, Q. C.; Guo, Y.; Zhao, P. X.; Lu, T. X.; Wan, X. L.; Rong, X. X.; Pan, W. (2017). Optimization and simulation for airport emergency inventory with replacement, *International Journal of Simulation Modelling*, Vol. 16, No. 1, 133-144, doi:[10.2507/IJSIMM16\(1\)CO1](https://doi.org/10.2507/IJSIMM16(1)CO1)
- [31] Katsios, D.; Xanthopoulos, A. S.; Koulouriotis, D. E.; Kiatipis, A. (2018). A simulation optimisation tool and its production/inventory control application, *International Journal of Simulation Modelling*, Vol. 17, No. 2, 257-270, doi:[10.2507/IJSIMM17\(2\)425](https://doi.org/10.2507/IJSIMM17(2)425)