Journal of Optimization in Industrial Engineering
Vol.14, Issue 2, Summer & Autumn 2021, 53-72
DOI:10.22094/JOIE.2020.1873983.1674

Research Article

Scheduling on Flexible Flow Shop with Cost-related Objective Function Considering Outsourcing Options

Mojtaba Enayati[a], Ebrahim Asadi-Gangraj[a], Mohammad Mahdi Paydar[a]

[a] *Babol Noshirvani University of Technology; Babol; Iran.*

Received 15 July 2019 ; Revised 12 October 2020 ; Accepted 25 October 2020

Abstract

This study considers outsourcing decisions in a flexible flow shop scheduling problem, in which each job can be processed by either an in-house production line or outsourced. The selected objective function aims to minimize the weighted sum of tardiness costs, in-house production costs, and outsourcing costs with respect to the jobs due date. The purpose of the problem is to select the jobs that must be processed in-house, schedule processing of the jobs in-house, and finally select and assign other jobs to the subcontractors. We develop a mixed-integer linear programming (MILP) model for the research problem. Regarding the complexity of the research problem, the MILP model cannot be used for large-scale problems. Therefore, four metaheuristic algorithms, including SA, GA, PSO, hybrid PSO-SA, are proposed to solve the problem. Furthermore, some random test problems with different sizes are generated to evaluate the effectiveness of the proposed MILP model and solution approaches. The obtained results demonstrate that the GA can obtain better solutions in comparison to the other algorithms.

Keywords: Flexible flow shop scheduling; Outsourcing; Cost-related objective functions; Metaheuristic algorithms

1. Introduction

Scheduling problem is a branch of the operations research that is widely studied during the last years. It is one of the main activities in some production and service systems (Hosseini, 2019). Scheduling aims to allocate finite resources to the jobs to optimize the objective functions (Behnamian, 2020). It refers to the problems in manufacturing systems, in which, the jobs must be scheduled to process on one or more machines concerning a wide range of the objective functions. Regarding the job features, machines layout in the production lines, processing constraints, and objective functions, there is a large variety of scheduling problems in the machine scheduling domain.

The flow shop scheduling problem is a common manufacturing system, so that it is considered by some researchers since Johnson's seminal paper (Johnson, 1954). In the flow shop problem, several stages are located in series, where there is only one machine in each stage (Nahavandi and Asadi-Gangraj, 2014). Nowadays, to increase the capacity, balance the capacities in a particular stage, and increasing demand for some products, it has been led to consider parallel machines in some stages in many companies. This developed environment is called as flexible flow shop (FFS), hybrid flow shop (HFS), flexible flow line (FFL), hybrid flow line (HFL), or flow shop with multiprocessor. In the FFS environment, machines are arranged into m stages in series, so that in stage i, there are S_i unrelated parallel machines (Asadi-Gangraj, 2018).

The FFS problem combines two types of well-known scheduling problems: parallel machine scheduling problem and flow shop scheduling problem. The purpose of the FFS is to specify two decisions: sequence of the jobs through the shop-floor and allocation of the jobs to machines. Hence, the FFS scheduling problem tries to determine assignment of the jobs to the machines and scheduling them in each stage.

Nowadays, most companies use outsourcing option in their industries, where they give some jobs to subcontractors. Via proper outsourcing decision, different costs of the company, such as operating costs, inventory costs, and delivering costs, can be reduced and it also leads to the company to be more flexible. With respect to the globalization and information technology growth, outsourcing plays a key role in manufacturing systems and gives some benefits to the companies in different manners. Through outsourcing some no-serious tasks to the subcontractors, the company can concentrate more on its main tasks.

In this paper, an FFS scheduling problem is studied where each job must be processed by in-house production line or subcontractor production line. The objective of the problem is to minimize tardiness costs, in-house production costs, and outsourcing costs. The main purpose of the problem is to make a subset of jobs that are associated with the in-house production line then scheduling these jobs, and make a subset of jobs that are associated with subcontractors and select a subcontractor for each outsourced job.

*Corresponding author Email address: e.asadi@nit.ac.ir

The remaining of the paper is structured as follows. A brief review of the related studies is provided in section 2. In Section 3, the proposed MILP model is described. Section 4 presents the proposed metaheuristic algorithms. Section 5 is devoted to provide computational experiments and finally, conclusions and future research are presented in section 6.

2. Literature Review

There are many studies in simple environments such as single machine and parallel machine environments about the scheduling problems with outsourcing options. Ruiz-Torres et al. (2006) presented a parallel machine scheduling problem with outsourcing option. The objective function minimizes outsourcing time and cost, and the number of tardy jobs. Lee and Sung (2008) dealt with a single machine scheduling problem by considering the outsourcing, where jobs must be either processed in-house or outsourced. The selected objective function is to minimize the weighted sum of the completion times and sum of the outsourcing cost. They also provided two heuristic approaches and a branch and bound (B&B) algorithm for this problem. Lee and Sung (2008) presented two scheduling problems in a single machine environment that outsourcing is allowed. The objective is to minimize maximum lateness and total tardiness considering outsourcing budget. Because of NP-hardness, they proposed two heuristic approaches and a B&B algorithm. Qi (2008) studied a scheduling problem with outsourcing in a single machine environment. There is a subcontractor with single machine environment and outsourced orders must ship back to the in-house shop in batches. The objective aims to minimize the weighted sum of makespan and total outsourcing and transportation cost. He proposed a dynamic programming approach for this problem. Chen and Li (2008) considered a parallel machine scheduling problem with outsourcing options with total production and outsourcing costs minimization. They proposed a heuristic algorithm for this problem. Mishra et al. (2008) developed an MILP model for the integrated planning and scheduling with respect to outsourcing supply chain. They presented a method to make strategic decisions. Chan et al. (2009) developed an enhanced swift converging simulated annealing algorithm for the scheduling problem with outsourcing option. The proposed algorithm is evaluated by comparing with other metaheuristic methods, GA, SA, TS and TS-SA algorithms. The results showed that this algorithm has a better performance to choose the best subcontractor for the jobs. Haoues et al. (2013) considered a scheduling problem to deal simultaneously with the in-house scheduling and outsourcing cost. They developed a GA-based algorithm to handle the research problem. Mokhtari and Abadi (2013) developed a mathematical model for planning the in-house and outsourced jobs, simultaneously. The selected objective function minimizes sum of the total weighted completion time and total outsourcing cost. Zhong and Huo (2013) presented a single machine scheduling problem with outsourcing option, in which each job can be processed within the in-house production line or to be outsourced to the subcontractors. They considered two objective functions for this problem. Neto et al. (2015) studied a parallel machine scheduling problem by considering the outsourcing option. They proposed an ant colony optimization algorithm to solve this problem. The selected objective function aims to minimize sum of the outsourcing and tardiness costs. Choi et al. (2016) presented a mini-max regret of a single machine scheduling problem to determine which jobs should be outsourced to the subcontractors with total in-house and outsourcing production costs minimization. They proposed heuristic algorithms to solve this problem.

Also, some studies are presented in a more complicated environment such as flow shop or job shop environments. Lee et al. (2002) developed a mathematical model for advanced planning and scheduling problem with respect to the outsourcing options. Also, they developed a GA-based heuristic approach to solve this problem. Chung et al. (2005) presented a job shop scheduling problem with outsourcing to satisfy the due dates. They developed a heuristic algorithm for this problem. Qi (2009) dealt with makespan minimization in a two-stage flow shop with outsourcing options, so that only some of the operations can be outsourced. Lee and Choi (2011) dealt with a two-stage flow shop scheduling problem, in which each job can be processed by using in-house production line or outsourced to the subcontractor. The selected objective minimizes the weighted sum of the makespan and the total outsourcing costs. Neto and Filho (2011) proposed two independent ant colony optimization algorithms for the scheduling problem in permutation flow shop environment with outsourcing with respect to minimizing the makespan and outsourcing costs. Choi and Chung (2011) studied a two-machine flow shop scheduling problem with outsourcing options in order to minimize makespan and total outsourcing costs. They developed a polynomial-time algorithm to solve the problem. Qi (2011) presented a two-stage flow shop scheduling problem with respect to options of outsourcing some operations to the subcontractors to minimize the makespan. They considered different modes of outsourcing and proposed optimization algorithms for each mode. Moghaddam et al. (2012) presented a mathematical model for a two-machine flow shop scheduling problem with outsourcing options. The objective function is to minimize the total completion time for the in-house jobs and outsourcing cost. They developed a genetic-based approach to solve large-size problems. Mokhtari et al. (2012) dealt with a multi-stage flow shop scheduling problem, where the outsourcing option is allowed. The objective of the problem is to minimize the sum of weighted flow time, transportation cost, and processing time of outsourced jobs. They also proposed an MILP model and a team process algorithm. Chung et al. (2013) studied a two-machine flow shop scheduling problem with the outsourcing option to minimize sum of the makespan and the total outsourcing cost. They developed an approximation algorithm for this problem. Guo et al. (2014) studied a bi-objective job shop scheduling problem considering the outsourcing, with respect to total outsourcing and tardiness cost. They used a lexicographic approach to consider these objectives, simultaneously, and proposed a two-phase neighborhood search to solve the problem. Lei et al. (2016) presented a job shop scheduling problem with outsourcing options. The objective function minimizes total tardiness regarding the limited outsourcing budget. They proposed a novel shuffled frog leaping algorithm for this problem. Ahmadizar and Amiri (2018) dealt with outsourcing in a two-machine flow shop scheduling problem to minimize the sum of the makespan and outsourcing and transportation costs. They presented two mathematical models and an ACO algorithm for this problem. Tirkolaee et al. (2020) studied outsourcing option and Just-in-Time delivery in the slow shop scheduling environment. They proposed a bi-objective MILP model and hybrid version of interactive fuzzy solution technique and a Self-Adaptive Artificial Fish Swarm Algorithm to minimize the total cost of the production system and total energy consumption. Wand and Cui (2020) considered robust identical parallel machine scheduling problem with outsourcing option and uncertain processing time. The selected objective function aims to minimize total completion time of in-house jobs and the cost of outsourcing jobs. They developed two approximation algorithms for the problem with discrete and interval scenarios.

In order to convenience the reader, some of the main researches in the context of scheduling problem with outsourcing are summarized in Table 1.

Table 1
A brief overview of the literature review

Year	**Author(s)**	**Environment**	**Objective Function**	**Solving Method**	**Number of subcontractors**
2002	Lee et al.	Flow shop	Makespan	GA-based approach	One
2005	Chung et al.	Job shop	Outsourcing cost	Heuristic algorithms	One
2006	Ruiz-Torres et al.	Parallel machine	Total external machine utilization and the total number of late jobs	Heuristic algorithms	One
2008	Chen and Li	Parallel machine	Production and outsourcing costs	Heuristic algorithms	Multiple
2009	Qi	Flow shop	Outsourcing cost and makespan	Heuristic algorithms	one
2009	Chan et al.	Parallel machine	Makespan	GA, SA, and Fuzzy Logic Controller (FLC)	one
2011	Lee and Choi	Flow shop	Makespan and outsourcing cost	Heuristic algorithms	one
2012	Moghaddam et al.	Flow shop	Makespan and outsourcing cost	GA-based algorithm	One
2012	Mokhtari et al.,	Flow shop	Flow time and outsourcing cost and transportation cost	Team process	Multiple
2013	Mokhtari and Abadi	Parallel machine	Completion time and outsourcing cost	Heuristic algorithms	Multiple
2013	Chung and Choi	Flow shop	Makespan and outsourcing cost	Heuristic algorithms	One
2014	Guo and Lei	Job shop	Total tardiness and outsourcing cost	Heuristic algorithms	One
2015	Neto et al.	Parallel machine	Sum of outsourcing and delay costs	Ant colony optimization	One
2016	Lei and Guo	Job shop	Tardiness and outsourcing cost	Shuffled frog-leaping algorithm	One
2016	Choi and Chung	Single machine	Total production cost	Heuristic algorithms	One
2018	Ahmadizar and Amiri	Flow shop	Makespan, transporting and outsourcing cost	ACO-based algorithm	Two
2020	Tirkolaee et al.	Flow shop	Total cost of the production and total energy consumption.	Hybrid version of interactive fuzzy solution technique and a Self-Adaptive Artificial Fish Swarm Algorithm	One
2020	Wang and Cui	Identical parallel machine	Total completion time of in-house jobs and the cost of outsourcing jobs	Two approximation algorithms	One
2020	Present research	Flexible flow shop	Tardiness costs, in-house production costs, outsourcing costs	SA, GA, PSO, hybrid PSO-SA	Multiple

2.1. Research gap and contributions

The presented study investigates the flexible flow shop scheduling problem with unrelated parallel machines concerning outsourcing option. Three cost related objective functions, including tardiness costs, in-house production costs, and outsourcing costs, are considered in this research. In order to tackle this problem, a mixed-integer linear programming model is introduced. Besides, regarding the NP-hardness of this problem, four metaheuristic approaches, namely SA, GA, PSO, hybrid PSO-SA are proposed to solve this problem. With respect to the pervious section and Table 1, the main contributions of this study are summarized in the following:

- The present research is the first try to consider the outsourcing option in the flexible flow shop environment with unrelated parallel machines.
- Multiple subcontractors are rarely dealt with in the scheduling problems with outsourcing options. As result, we consider this issue in the present research.
- Most of the papers in this context are used heuristic approaches for the scheduling problem with outsourcing option. Hence, four metaheuristic approaches are proposed to tackle the research problem, in which, a new heuristic approach is applied in the body of the metaheuristics to 1) divide the jobs into two subsets, including in-house jobs and outsourced jobs, 2) schedule the jobs in the in-house production line, 3) assign the outsourced jobs to the subcontractors, and finally 4) schedule the jobs on the subcontractor.
- A wide range of objective functions, including tardiness costs, in-house production costs, and outsourcing costs is considered in this paper.

3. Problem Description

There are n different jobs $(j = 1, \dots, n)$, in which each job must be processed on in-house production line or outsourced. In-house machine layout is considered as flexible flow shop, in which, there some machines, which are arranged into s stages in series and there are k $(k = 1, \dots, m_i)$ unrelated machines in parallel in each stage. Each machine can process only one job at a time and each job can be processed by only one machine at a time. Entire jobs have to be processed on only one machine at each stage and they are available at time zero. The travel time is neglected between the stages and setup time is considered in the processing times. Because of unrelated parallel machines at each stage, jobs processing time are different with respect to the different machines. Preemption is not allowed and there is unlimited storage space between the two successive stages.

Also, there are h, $(h = 1, \dots, H)$ different subcontractors, where each job that selected for outsourcing is fully outsourced to them. Each outsourced job is processed by one of the available subcontractors. The objective is to find an integrated schedule for the in-house and outsourced jobs to minimize the tardiness costs, in-house production costs, and outsourcing costs.

A schematic of the research problem is presented in Figure 1.

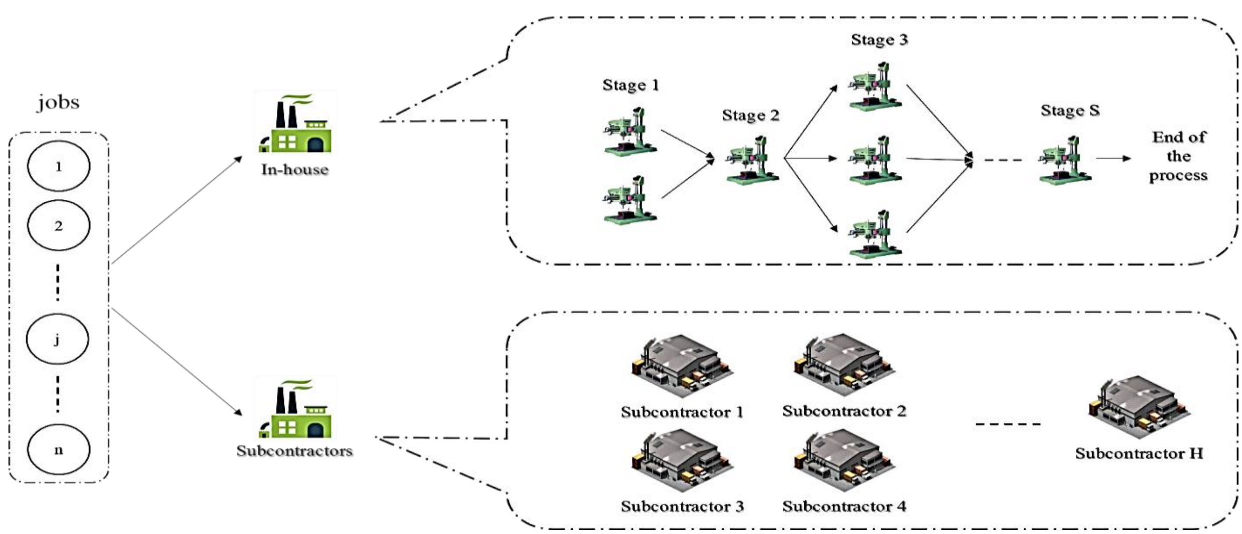

Fig. 1. A schematic of the research problem

3.1. Mathematical model

In this section, we will introduce parameters and decision variables to propose the mixed-integer linear programming (MILP) model for the research problem.

Parameters

The necessary parameters to present the proposed mathematical model are as follows:

s: Number of stages
n: Number of jobs
H: Number of subcontractors
i: Stage index
j, l: Job index
h: Subcontractor index, $h \in (1, \dots, H+1)$.
$H+1$: Index of in-house production environment
m_i: Number of machines in stage i
P_{ikj}: Processing time of job j on machine k in stage i
IC_j: In-house production cost of job j
t_h: Transportation time from in-house production line to subcontractor h
PO_{hj}: Processing time of job j on subcontractor h
TC_j: Tardiness cost of job j
OC_{hj}: Outsourcing cost of job j on subcontractor h
M: A big number

Decision variables

T_j: Tardiness of job j
C_{ij}: Completion time of job j on stage i
CO_{hj}: Completion time of job j on subcontractor h.
X_{ikj}: if job j at stage i is assigned to machine k, 1, otherwise, 0.
Y_{ijl}: if job l is processed after job j at stage i, 1, otherwise, 0.
Z_{hj}: if job j assign to subcontractor h, 1, 0, otherwise. If job j assign to in-house shop, $z_{H+1} = 1$.
YO_{ljh}: if job j is processed after job l on subcontractor h, 1, otherwise, 0.

MILP model

The formulation of the MILP problem is as follows:

$$Min\ Z = \sum_j TC_j T_j + \sum_{h \neq H+1} \sum_j OC_{hj} Z_{hj} + \sum_j IC_j Z_{H+1,j} \quad (1)$$

$$C_{1j} \geq \sum_{k=1}^{m_1} P_{1kj} X_{1kj} \qquad \forall j \quad (2)$$

$$C_{ij} \geq C_{i-1,j} + \sum_{k=1}^{m_i} P_{ikj} X_{ikj} \qquad \forall i \geq 2, j \quad (3)$$

$$C_{il} \geq C_{ij} + P_{ikl} X_{ikl} - M(3 - X_{ikj} - X_{ikl} - Y_{ilj}) \qquad \forall i, j \neq l, k \quad (4)$$

$$C_{ij} \geq C_{il} + P_{ikj} X_{ikj} - M(2 - X_{ikj} - X_{ikl} + Y_{ilj}) \qquad \forall i, j \neq l, k \quad (5)$$

$$C_{ij} \leq C_{i-1,j} + M X_{H+1,j} \qquad \forall i, j \quad (6)$$

$$\sum_{h=1}^{H+1} Z_{hj} = 1 \qquad \forall j \quad (7)$$

$$\sum_{k=1}^{m_i} X_{ikj} = Z_{H+1,j} \qquad \forall i, j \quad (8)$$

$$CO_{hj} \geq \sum_{h=1}^{H} (PO_{hj} + t_h) Z_{hj} \qquad \forall j, h < H+1 \quad (9)$$

$$CO_{hj} \geq CO_{hl} + PO_{hj} Z_{hj} - M(3 - Z_{hj} - Z_{hl} - YO_{ljh}) \qquad \forall j, h < H+1 \quad (10)$$

$$CO_{hl} \geq CO_{hj} + PO_{hl} Z_{hl} - M(2 - Z_{hj} - Z_{hl} + YO_{ljh}) \qquad \forall j, h < H+1 \quad (11)$$

$$CO_{hj} \leq M Z_{hj} \qquad \forall j, h < H+1 \quad (12)$$

$$T_j \geq C_{Sj} - d_j \qquad \forall j \quad (13)$$

$$T_j \geq CO_{hj} - d_j \qquad \forall j, h < H+1 \quad (14)$$

$$T_j \geq 0, C_{ij} \geq 0, CO_{hj} \geq 0 \qquad \forall i, j, h \qquad (15)$$

$$X_{ikj}, Y_{ilj}, Z_{hj}, Z_{hj} \in \{0,1\} \qquad \forall i, k, j, l, h \qquad (16)$$

The objective function (1) aims to minimize the total costs, including total tardiness costs, in-house production costs, and outsourcing costs. Constraint set (2) determines the completion time of in-house jobs in the first stage and constraint set (3) calculates the completion time of in-house jobs in the other stages. Constraint sets (4) and (5) preclude the interference between the processing operations of any two jobs, which are processed on one machine in each stage. In a moment, at most one of these two constraint sets is activated. If jobs j and l are processed on machine k in stage i ($X_{ikj} = X_{ikl} = 1$) and j is sequenced before l ($Y_{ilj} = 1$), constraint set 4 is activated. On the other side, if job l is sequenced before j ($Y_{ilj} = 0$), constraint set 5 will be activated. Finally, if jobs j and l are sequenced on different machines ($X_{ikj} + X_{ikl} \leq 1$), both constraint sets will be redundant. Constraint set (6) ensures that in-house completion times of the outsourced jobs must be equal to 0. Constraint set (7) enforces that each job must be processed in an in-house production line or outsourced to one of the subcontractors. Constraints set (8) indicates that in-house jobs must be assigned to only one machine at each stage. Constraints set (9) calculates completion time of outsourced jobs regarding the transportation time from in-house production line to the subcontractor and processing time of the subcontractor. Constraint sets (10) and (11) consider the completion time of jobs j and l, if they are outsourced to a subcontractor. Constraints set (12) indicates that completion times of in-house jobs on the subcontractors equal to 0. Constraint sets (13) and (14) determine tardiness of the in-house and outsourced jobs, respectively. Finally, constraint sets (15) and (16) show the range of decision variables.

4. Solutions Approaches

Since the flexible flow shop problem with unrelated machines is NP-hard (Aadi-Gangraj, 2018) in strong sense, the flexible flow shop problem with outsourcing option is also NP-hard. Therefore, the problems with medium to large size cannot be solved through the exact methods in a reasonable time; hence, in this study, we applied four metaheuristic approaches. For this purpose, four metaheuristic approaches, including simulated annealing (SA), genetic algorithm (GA), particle swarm optimization algorithm (PSO) and a hybrid version of SA and PSO (PSO-SA) algorithm, will be introduced.

4.1. Simulated annealing algorithm

SA algorithm is firstly introduced by Kirkpatrick, Gelatt, and Vecchi (1983) and Cerny (1985) and it is one of the frequently used algorithms in the optimization problems. This approach is inspired by a procedure that includes heating and controlled cooling of a material to get the bigger size of its crystals and decrease their fault (Nayeri, et al. 2019). In order to solve an optimization problem, the SA algorithm first starts with an initial solution, and then moves to the neighboring solutions in a repeating loop. If the new solution is better than the current solution, the algorithm selects it as the best solution. Otherwise, the algorithm accepts the new solution with the probability $exp^{(-\Delta E/T)}$ (Boltzmann operator) as the best solution. In this formula, ΔE shows the difference between the objective function of new solution and the current solution, and T is the current temperature. Several iterations are performed in each temperature and the temperature is reduced, slowly. This process is performed until the stopping criteria have met. Regarding the literature, the SA algorithm has been successfully applied to a wide range of the scheduling problems. Following, different components of the SA algorithm is introduced.

4.1.1. Solution structure

The first step in solving a problem with metaheuristic methods is to create an appropriate solution structure. The initial solution structure which is used in this algorithm is a vector in size of *1*(number of jobs)*, so that each number represents a job and the vector presents a sequence of jobs.

4.1.2. Create a neighborhood solution

In the SA algorithm, different methods are applied to create neighborhood solutions, such as swap, reversion, and insertion. In this research, by using a roulette wheel, these methods have been applied, randomly.

Swap: In the swap method, two genes of the sequence vector are selected randomly and the corresponding jobs are swapped. Figure 2 depicts an example of the swap with nine jobs.

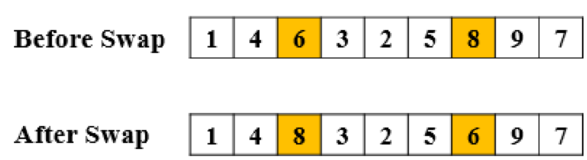

Fig. 2. An example of swap

Reversion: In the reversion method, by selecting two genes of the chromosome, the sequence of genes between them is reversed. An example of the revision method is showed in Figure 3.

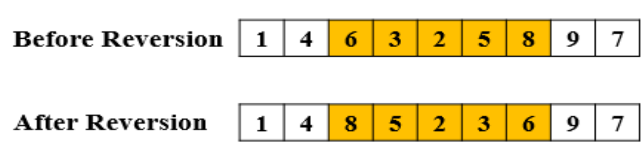

Fig. 3. An example of reversion

Insertion: Besides the aforementioned method, in the insertion method, two genes are firstly selected, and the second gene is removed and added after the first gene. This method can be seen in Figure 4.

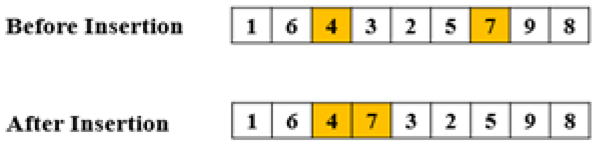

Fig. 4. An example of insertion

4.2. Genetic Algorithm (GA)

Genetic algorithm (Goldberg and Holland, 1988) is an iterative algorithm such that natural evolution is applied to model the search method. GA is one of the most widely used metaheuristic approaches for different optimization problems.

At the beginning of the GA, a random population is created and evaluated. Then, a percentage of the population (chromosome) is selected as the parent and children's population are formed by the combination of them. Similarly, a percentage of the population is selected for the mutation operations and the mutated population is generated. In the next step, the main population, children and mutated population are merged and the new main population is generated. If the stopping condition is fulfilled, the algorithm ends. Otherwise, the process will be repeated.

4.2.1 Solution structure

One of the main steps of the GA is to represent the solutions as a chromosome (Rezaeian and Zarook, 2018). The chromosome is used to represent the solution that is similar to the one which is introduced for the SA algorithm. In order to correctly apply the operators of the GA on the chromosome, random numbers are generated in the interval of [0,1] and assign to each cell. Then, the numbers in the sequence vector are ranked and entered in the same vector. An example of the sequence vector for nine jobs is showed in Figure 5.

0.25	**0.58**	**0.93**	**0.47**	**0.16**	**0.64**	**0.88**	**0.74**	**0.36**
2	**5**	**9**	**4**	**1**	**6**	**8**	**7**	**3**

Fig. 5. An example of sequence vector in the GA

4.2.2 GA operators

As mentioned before, the next generation of the population is generated with respect to crossover and mutation operators.

Crossover: The genetic combination of two or more chromosomes to produce the children for the next generation is called crossover operation. In the present research, three types of the crossover are used: single-point crossover, uniform crossover, and double-point crossover. Figure 6 illustrates the aforementioned crossover.

Table 1
Parameters and their values

Algorithm	Parameter	Description	Legend	Level 1	Level 2	Level 3
SA	MaxIt	Maximum iteration	A	50	100	150
	MaxSubIt	Maximum of sub-iteration	B	10	20	30
	T	Initial temperature	C	500	700	1000
	Alpha	Temperature reduction rate	D	0.7	0.8	0.9
GA	MaxIt	Maximum iteration	A	50	100	150
	Popsize	Population size	B	10	20	30
	Cross Rate	Crossover rate	C	0.7	0.8	0.9
	Mut Rate	Mutation rate	D	0.3	0.2	0.1
PSO	MaxIt	Maximum iteration	A	50	100	150
	PopSize	Population size	B	10	20	30
	c_1	perceptual factor	C	0.75	0.85	0.95
	c_2	Social factors	D	0.95	0.85	0.75
PSO-SA	MaxIt	Maximum iteration	A	50	100	150
	PopSize	Population size	B	10	20	30
	c_1	perceptual factor	C	0.75	0.85	0.95
	c_2	Social factors	S	0.95	0.85	0.75
	T	Initial temperature	E	500	700	1000
	Alpha	Temperature reduction rate	F	0.7	0.8	0.9

Signal-to-noise (*S/N*) ratio is used to select the best levels of the metaheuristic parameters. The signal-to-noise graphs for the experiments are shown in Figures 8-11.

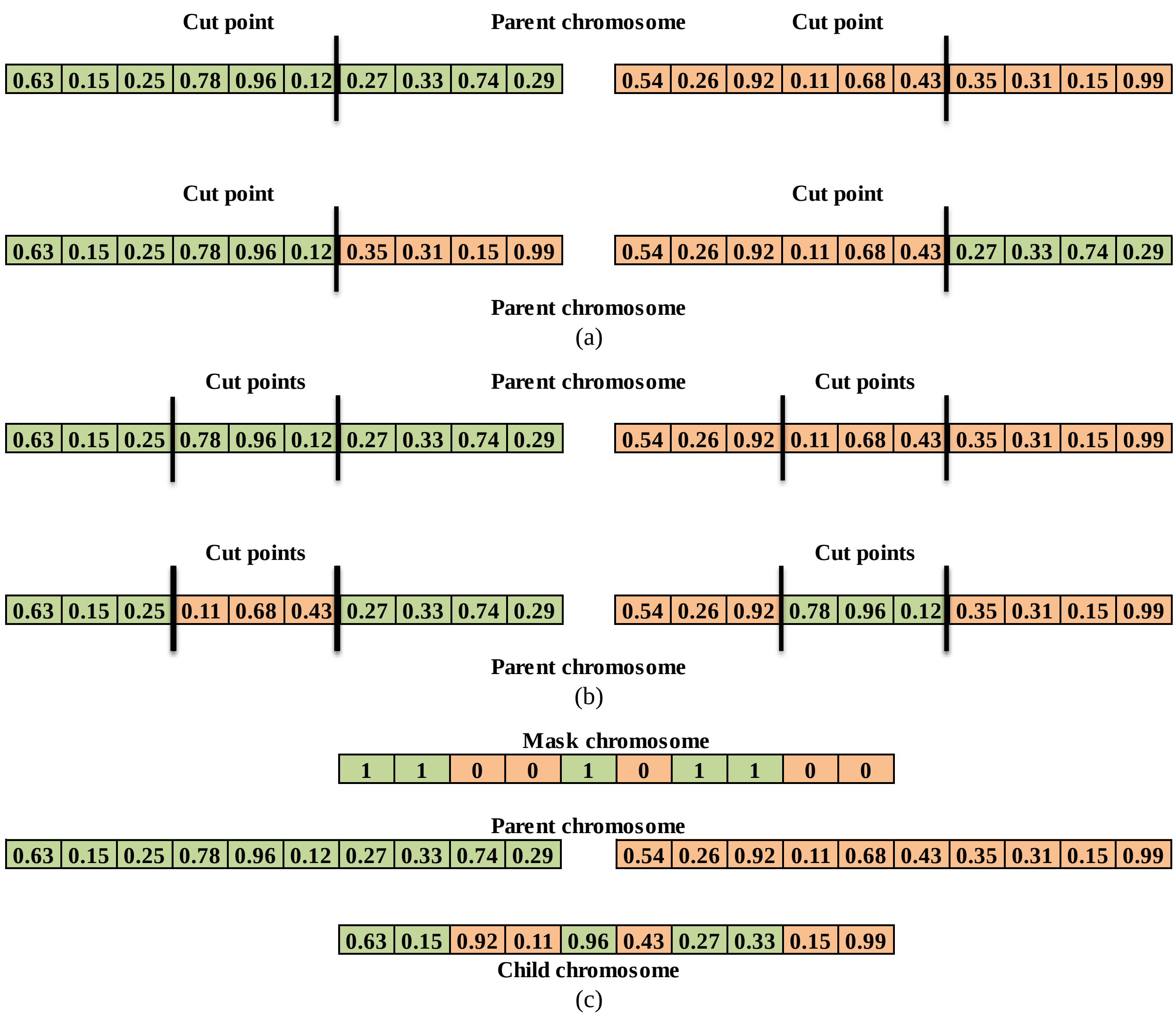

Fig. 6. Different crossover operators; (a) Single-point crossover; (b) Double-point crossover; (c) Uniform crossover

Mutation: In nature with a small possibility, a child has a characteristic that is not a genetic characteristic of his/her parents. This feature is considered as a mutation in the GA algorithm. In this research, one of the parent's genes is chosen randomly, and the value of that gene is randomly changed.

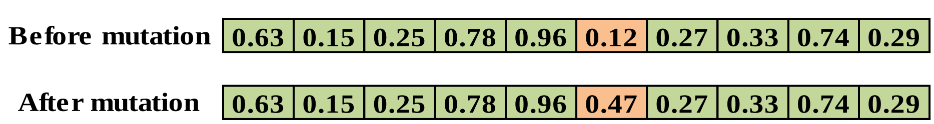

Fig. 7. An example of mutation operator

4.3. Particle Swarm Optimization (PSO)

PSO algorithm is a population-based algorithm which is firstly introduced by Eberhart and Kennedy in 1995. It coincides with the simulation of the social behavior of the animals such as birds, bees, and fish. In order to simulate the search for the food, the members determine their speed based on two factors: their own best experience and population best experience. Each member in the group (particle) is defined by position and velocity so that the particle position represents the solution for the optimization problem and the velocity shows the direction and distance to guide the movements of the particles. Each particle determines its movement based on its own experience and its neighbors to find the optimal/near optimal solution. Each particle also updates the velocity with respect to the current velocity, the best position by the particle (*pbest*), and the best position experienced by all particles (*gbest*). In each iteration, the particle velocity and position are updated as follows:

$$Vel_i(k+1) = w \times Vel_i(k) + c_1 \times r_1 \times \left(pbest_i - X_i(k)\right) + c_2 \times r_2 \times (gbest - X_i(k)) \tag{17}$$

$$X_i(k+1) = X_i(k) + Vel_i(k+1) \tag{18}$$

In which, $Vel_i(k)$ is the velocity of particle i in iteration k and $X_i(k)$ indicates position of particle i in iteration k. $pbest_i$ is the best position of particle i and $gbest$ is the best position ever obtained. r_1 and r_2 are random numbers between (0,1), c_1 and c_2 are perceptual and social factors, respectively, and w is the inertial factor.

4.3.1 Solution structure

The solution structure is used to represent the solution is similar to the one which is introduced for the SA algorithm.

4.4. Hybrid PSO-SA algorithm

This section is devoted to introduce a hybrid version of the SA and PSO algorithms to solve the FFS problem with outsourcing options. In this approach, the initial population is produced regarding the proposed method which is introduced in the last section. The Boltzmann operator in the SA algorithm is also applied to update the $pbest$. If the objective function of the new solution is better than the last $pbest$, the $pbest$ is updated, otherwise the new solution will be accepted as new $pbest$ regarding the obtained probability by the Boltzmann probability function. Other details of this algorithm are the same as the PSO algorithm.

4.5. Calculation the objective function

In all the metaheuristic algorithms, which are explained in previous sections, the objective function is calculated as follows:

First of all, the initial sequence of the jobs is generated regarding sequence vector. Then, entire jobs are processed in the in-house production line. The assigning process to the machine in each stage is as follows. Each job is temporarily assigned to all the available and unavailable machines in each stage. This may be due to the fact that, regarding the unrelated parallel machines at each stage, an unavailable but more efficient machine may produce an earlier completion time for the job. Then, a machine with minimum completion time is selected and the job is assigned to this machine, permanently. This approach is continued until entire jobs are assigned to the machine in each stage. Then the objective function, including tardiness cost and in-house production cost, is calculated.

In the next step, the outsourced jobs must be determined. The last job in the sequence vector is removed from the in-house production line and assigned to any subcontractor. Then, the objective function is approximately estimated. It is due this fact that deleting a job from the in-house production line and assigning it to the subcontractor may lead to changing the tardiness cost of the other jobs. Besides, exact calculation of the objective function for any sequence is very time-consuming. Thus, the approximate estimation of the objective function is taken into account in this stage. If assigning the job to the subcontractor is led to the decreasing the objective value, the exact objective value is calculated and the job is permanently assigned to the corresponding subcontractor. This process is repeated for entire jobs in the sequence. Eventually, tardiness cost, in-house production costs, and outsourced production costs are determined regarding to the subsets of the in-house and outsourced jobs.

5. Experimental Results

This section is devoted to investigating the performance of the proposed approaches for the flexible flow shop scheduling problem with outsourcing options. First of all, some experiments are conducted to tune the metaheuristics parameters. Then, some random test problems are generated in different sizes to analyze the performance of the metaheuristic approaches.

5.1. Parameters Setting

It is obvious that the performance of the metaheuristic algorithms depends on its parameters. As result, we applied Taguchi method to tune the parameters. The levels of the parameters for all the metaheuristic algorithms and their description, legend, and values, are provided in Table 2. For example, four parameters, including maximum of iterations (MaxIt), maximum of sub-iteration (MaxSubIt), initial temperature (T), and temperature reduction rate (alpha) are considered for the simulated annealing algorithm with three levels.

Fig. 8. The signal to noise graph for the SA algorithm

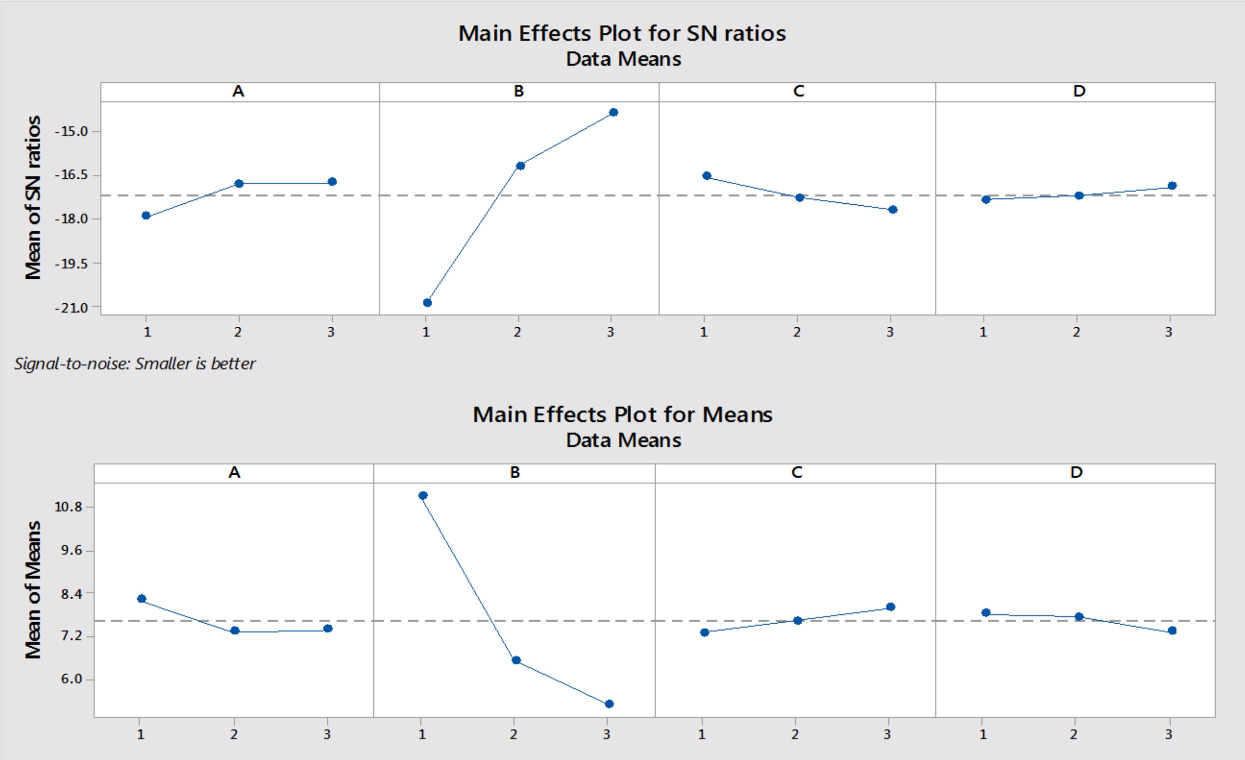

Fig. 9. The signal to noise graph for the GA algorithm

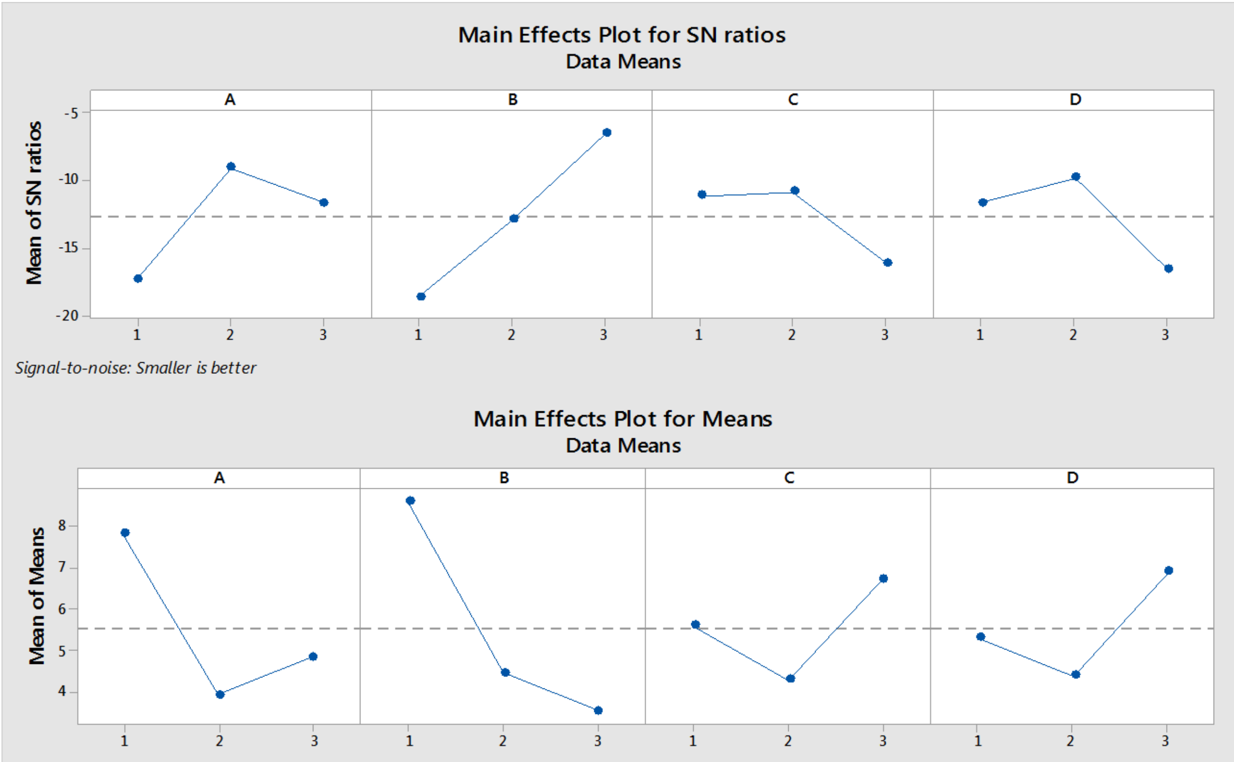

Fig. 10. The signal to noise graph for the PSO algorithm

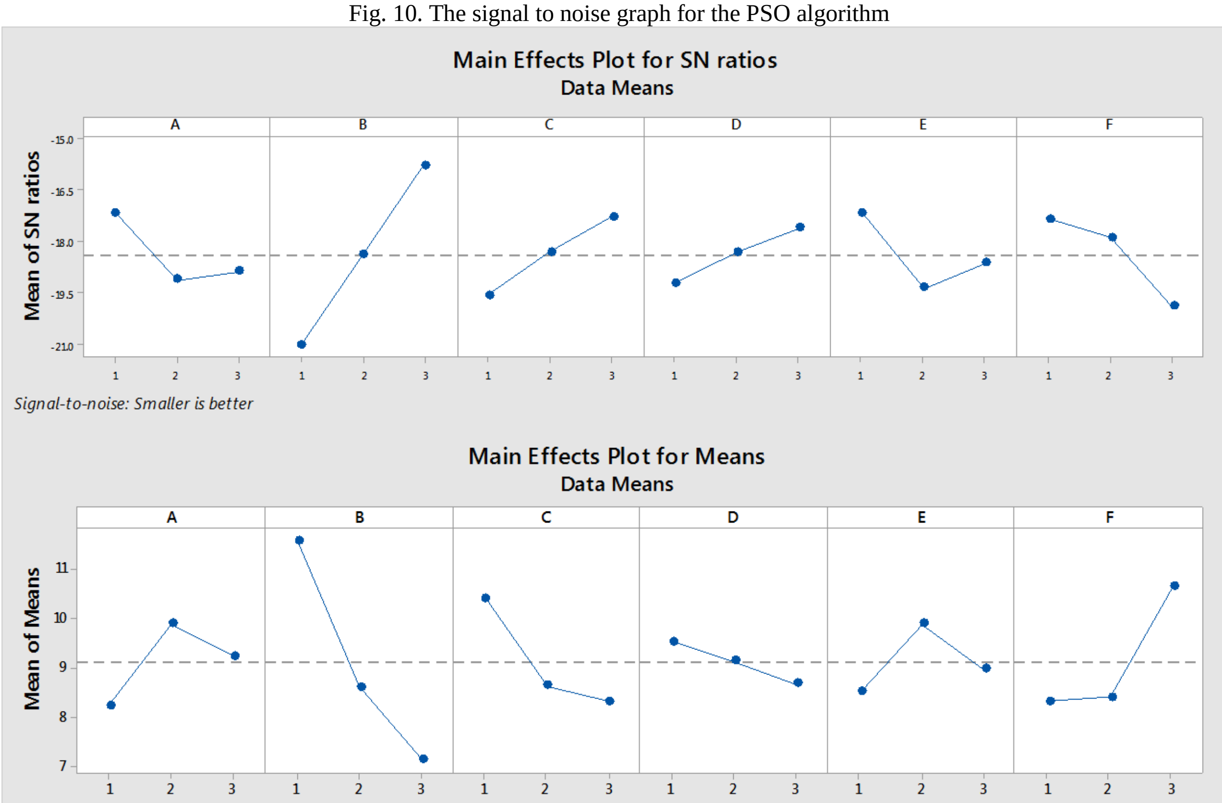

Fig. 11. The signal to noise graph for the PSO-SA algorithm

As results, the best values of the parameters are summarized in Table 3.

Table 2
The best value of the parameters for the metaheuristic algorithms

Algorithm	Parameter	Value
SA	MaxIt	150
	MaxSubIt	30
	T	500
	Alpha	0.8
GA	MaxIt	100
	Popsize	30
	Cross Rate	0.7
	Mut Rate	0.1
PSO	MaxIt	100
	Swarm-Size	30
	c_1	0.85
	c_2	0.75
Hybrid PSO-SA	MaxIt	50
	Swarm-size	30
	c_1	0.95
	c_2	0.85
	T	500
	Alpha	0.7

5.2. *Numerical experiments and results for the small and large-size test problems*

This section analyzes the performance of the proposed metaheuristic algorithms for the FFS scheduling problem with outsourcing options. For this purpose, two series of the random test problems, small-size test problem and medium to large-size test problems, are generated.

For small size test problems, 25 test problems with respect to Table 4 are generated and the obtained results by the SA, GA, PSO and PSO-SA algorithms are compared with the optimal result.

Table 3
Parameters of the small size test problems

Parameter	Range of parameter
Number of jobs	[4-12]
Number of stages	[2-10]
Number of subcontractors	[1-4]
Number of machines in each stage	[1-4]
In-house processing time	Uniform (10,30)
Contractor processing time	Uniform (50,100)
Transportation time	Uniform (20,50)

The optimal gap (OG) has been used to evaluate the proposed algorithms for the small-size test problems through expression (19):

$$OG = \frac{Metaheuristic\ Obj - Optimal\ Obj}{Optimal\ Obj} \qquad (19)$$

Where, $Metaheuristic\,Obj$ and $Optimal\,Obj$ are the optimum value and the objective value obtained by each metaheuristic algorithm, respectively. The experimental results are illustrated in Table 5. It is necessary to mention that the Lingo solver is interrupted after 7200 seconds.

Table 4
The performance of the proposed algorithms for small-scale test problems

Problem	Number of jobs	Number of stages	Number of subcontractors	Lingo		SA			GA			PSO			PSO-SA		
				Obj	Cpu time	Obj	OG	Cpu time	Obj	OG	Cpu time	Obj	OG	Cpu time	Obj	OG	Cpu time
1	5	3	1	2456	348	2527	0.03	2	2522	0.03	0.3	2485	0.01	0.3	2530	0.03	13
2	6	2	2	1875	2157	1906	0.02	2	1906	0.02	0.4	1907	0.02	0.3	1918	0.02	11
3	7	4	3	3798	3250	3808	0.00	14	3885	0.02	1	3809	0.00	3	3849	0.01	20
4	8	5	1	4332	---	4455	---	24	4400	---	4	4408	---	38	4370	---	3
5	8	5	4	3314	2362	3322	0.00	24	3377	0.02	4	3315	0.00	8	3336	0.01	1
6	4	3	1	3053	725	3072	0.01	0.6	3054	0.00	0.3	3078	0.01	0.3	3117	0.02	1
7	5	3	3	2702	130	2713	0.00	0.9	2705	0.00	0.4	2736	0.01	0.3	2738	0.01	2
8	10	3	3	3587	---	3664	---	30	3635	---	11	3615	---	64	3603	---	2
9	8	3	3	3456	4200	3457	0.00	13	3538	0.02	3	3537	0.02	0.3	3541	0.02	14
10	8	4	3	3591	3738	3679	0.02	11	3653	0.02	5	3620	0.01	2	3656	0.02	1
11	8	3	2	3367	3968	3420	0.02	20	3384	0.01	5	3467	0.03	10	3431	0.02	2

12	8	3	1	3970	1068	4074	0.03	3	3999	0.01	4	4037	0.02	0.6	4086	0.03	7
13	8	3	3	2768	2768	2842	0.03	1	2769	0.00	1	2796	0.01	1	2833	0.02	5
14	5	10	1	5476	117	5592	0.02	1	5604	0.02	0.4	5611	0.02	34	5553	0.01	0.3
15	5	10	1	3754	458	3844	0.02	1	3761	0.00	3	3801	0.01	0.6	3841	0.02	0.9
16	5	10	1	3201	576	3218	0.01	1	3212	0.00	1	3234	0.01	1	3244	0.01	5
17	6	10	1	3487	2587	3588	0.03	1.5	3516	0.01	1	3586	0.03	1	3539	0.01	23
18	8	10	1	3917	---	3925	---	7	4012	---	2	3917	---	2	4019	---	38
19	8	10	4	2637	1336	2679	0.02	3	2685	0.02	7	2700	0.02	2	2654	0.01	8
20	8	10	4	2368	---	2404	---	25	2405	---	5	2396	---	6	2403	---	0.3
21	4	5	1	3268	1024	3303	0.01	0.3	3340	0.02	0.3	3332	0.02	0.2	3322	0.02	0.3
22	6	5	2	2219	---	2223	---	1	2261	---	0.9	2283	---	0.8	2265	---	64
23	8	5	2	3068	---	3153	---	4	3151	---	5	3133	---	1	3110	---	0.3
24	10	5	3	2682	---	2722	---	2	2719	---	23	2734	---	1	2686	---	2
25	12	5	3	3022	2235	3062	0.01	2	3105	0.03	0.3	3024	0.00	1	3023	0.00	1

As can be seen in Table 5, the Lingo solver cannot attain the optimal solution for seven test problems. For the other test problems and based on the average OG ($\overline{OG}$) criteria, the GA (1.4%), PSO (1.4%), PSO-SA (1.6%), and SA (1.6%) have the best performance, respectively. Besides, Regarding the CPU time, the GA and the PSO-SA are the best and the worst algorithms, respectively. Also, the GA can achieve the optimal solution for 5 out of 18 test problems. As results, we can conclude that all the metaheuristic algorithms have good performance to achieve the optimal/near optimal solutions but the GA has the better performance.

Since the flexible flow shop scheduling problem with unrelated parallel machines is NP-hard, the FFS problem with outsourcing option is also NP-hard. Therefore, Lingo solver cannot achieve the optimal solutions for medium to large-size test problems in a reasonable time. As result, we only compare the metaheuristic algorithms with each other in this section. For this purpose, 100 test problems with random size are generated based on different parameters, including number of jobs, number of stages, number of subcontractors, number of machines in each stage, in-house processing time, transportation time to the subcontractors, subcontractor processing time. The values of the parameters are summarized in Table 6.

Table 5

Parameters of the large-size test problems

Parameters	Range
Number of jobs	(20,50,100)
Number of stages	(5,7,9)
Number of subcontractors	(3,4,5)
Number of machines in each stage	(3,5,7)
In-house Processing time	Uniform (10,30)
Subcontractor processing time	Uniform (50,100)
Transportation time	Uniform (20,50)

To solve test problems, the Matlab 2016 software runs on a system with AMD A8-7100 Radeon R5 1.8 GHz processor and 4GB of RAM. The percentage relative error (PRE) has been applied to evaluate the performance of the proposed metaheuristic algorithms.

$$RPE = \frac{Metaheuristic\ Obj - Best\ Obj}{Best\ Obj} \quad (20)$$

In which, $Metaheuristic\ Obj$ represents the objective function of each metaheuristic algorithm and $Best\ Obj$ shows the best objective function which is generated by the metaheuristic approaches. The obtained results are presented in Table 7.

Table 6
The performance of the proposed algorithms for large-size test problems

Problem	Number of jobs	Number of stages	Number of subcontractors	SA			GA			PSO			PSO-SA		
				Objective function	RPE	CPU	Objective function	RPE	CPU	Objective function	RPE	CPU	Objective function	RPE	CPU
1	20	5	3	50162	1.7%	20	49335	0.0%	26	50663	2.7%	26	50102	1.6%	20
2	50	5	4	338365	0.8%	150	335639	0.0%	162	336320	0.2%	142	346608	3.3%	172
3	100	5	5	111253	0.9%	303	110302	0.0%	345	112847	2.3%	356	110284	0.0%	383
4	20	7	3	12498	4.6%	73	12350	3.3%	65	11954	0.0%	68	12233	2.3%	83
5	50	7	4	50010	1.3%	211	49364	0.0%	195	50140	1.6%	225	49685	0.7%	204
6	100	7	5	120054	1.3%	549	118987	0.4%	558	118478	0.0%	554	119548	0.9%	558
7	20	9	3	20811	3.9%	211	20032	0.0%	265	20254	1.1%	224	20521	2.4%	324
8	50	9	4	58012	0.8%	433	57564	0.0%	425	57819	0.4%	452	58124	1.0%	452
9	100	9	5	98829	3.0%	835	99245	3.4%	884	95987	0.0%	846	96875	0.9%	932
10	20	5	3	17005	0.7%	27	16879	0.0%	30	17548	4.0%	25	16985	0.6%	31
11	50	5	4	36698	2.8%	184	36945	3.5%	192	35689	0.0%	235	36258	1.6%	235
12	100	5	5	92031	2.2%	340	90025	0.0%	356	90215	0.2%	345	91548	1.7%	359
13	20	7	3	21530	2.6%	98	21458	2.2%	86	21085	0.5%	98	20987	0.0%	84
14	50	7	4	51326	2.8%	256	49919	0.0%	298	50154	0.5%	245	51045	2.3%	275
15	100	7	5	141645	1.0%	531	142359	1.5%	536	140254	0.0%	530	142569	1.7%	605
16	20	9	3	20901	2.7%	207	20354	0.0%	235	21045	3.4%	204	20860	2.5%	231
17	50	9	4	51194	1.4%	435	50487	0.0%	436	52415	3.8%	432	50789	0.6%	425
18	100	9	5	116784	1.1%	898	116987	1.3%	885	115468	0.0%	904	117845	2.1%	912
19	20	5	3	13685	3.3%	22	13320	0.6%	19	13245	0.0%	23	13478	1.8%	24
20	50	5	4	44586	2.3%	145	43568	0.0%	156	45678	4.8%	184	44687	2.6%	196
21	100	5	5	87571	0.0%	386	88547	1.1%	394	89654	2.4%	325	88546	1.1%	404
22	20	7	3	17144	0.8%	165	17016	0.0%	158	17854	4.9%	135	17175	0.9%	158
23	50	7	4	44511	2.1%	258	43651	0.2%	221	44884	3.0%	221	43578	0.0%	221
24	100	7	5	97550	2.6%	538	95119	0.0%	584	98546	3.6%	589	95487	0.4%	606
25	20	9	3	19002	1.3%	225	18754	0.0%	224	18978	1.2%	212	18765	0.1%	245
26	50	9	4	58685	0.0%	585	59876	2.0%	586	60124	2.5%	545	59876	2.0%	556
27	100	9	5	148781	2.7%	932	145273	0.3%	975	150245	3.7%	1035	144875	0.0%	1056
28	20	5	3	21152	5.0%	26	20154	0.0%	26	20602	2.2%	28	20481	1.6%	29
29	50	5	4	44801	0.0	114	45214	0.9	124	45789	2.2	114	46578	4.0	125

					%			%			%			%	
30	100	5	5	122195	2.3 %	318	119457	0.0 %	300	123546	3.4 %	304	124587	4.3 %	384
31	20	7	3	14927	2.5 %	75	14568	0.0 %	75	14803	1.6 %	78	15076	3.5 %	74
32	50	7	4	56564	3.7 %	255	55486	1.7 %	256	55208	1.2 %	278	54568	0.0 %	248
33	100	7	5	114829	0.2 %	438	115452	0.8 %	414	115632	0.9 %	470	114587	0.0 %	485
34	20	9	3	18236	2.2 %	235	17845	0.0 %	215	18141	1.7 %	235	18440	3.3 %	254
35	50	9	4	46863	0.0 %	542	46937	0.2 %	580	47421	1.2 %	660	46987	0.3 %	580
36	100	9	5	110639	0.9 %	838	109684	0.0 %	808	112544	2.6 %	896	110633	0.9 %	880
37	20	5	3	12342	1.5 %	14	12157	0.0 %	12	12546	3.2 %	14	12754	4.9 %	11
38	50	5	4	35346	0.0 %	186	35506	0.5 %	187	36231	2.5 %	184	36108	2.2 %	175
39	100	5	5	89420	1.1 %	312	88927	0.5 %	302	88451	0.0 %	351	91045	2.9 %	376
40	20	7	3	25110	2.4 %	87	24519	0.0 %	84	25689	4.8 %	78	24655	0.6 %	95
41	50	7	4	64923	7.8 %	250	60215	0.0 %	270	61475	2.1 %	258	63542	5.5 %	268
42	100	7	5	158722	2.9 %	532	154258	0.0 %	565	154789	0.3 %	521	155421	0.8 %	602
43	20	9	3	19393	4.0 %	189	18845	1.1 %	198	18645	0.0 %	178	19275	3.4 %	198
44	50	9	4	47419	3.4 %	447	45876	0.0 %	456	47956	4.5 %	495	48654	6.1 %	435
45	100	9	5	120546	0.8 %	859	121457	1.6 %	884	119548	0.0 %	909	120124	0.5 %	941
46	20	5	3	17327	0.7 %	12	17209	0.0 %	14	17854	3.7 %	11	17744	3.1 %	12
47	50	5	4	39411	3.7 %	183	38956	2.5 %	176	38002	0.0 %	195	40524	6.6 %	175
48	100	5	5	102458	2.3 %	414	100175	0.0 %	425	101369	1.2 %	414	100361	0.2 %	432
49	20	7	3	15802	2.0 %	220	15485	0.0 %	214	16142	4.2 %	235	15831	2.2 %	214
50	50	7	4	41996	1.1 %	254	41687	0.4 %	264	42755	2.9 %	242	41534	0.0 %	245
51	100	7	5	88956	1.7 %	553	87456	0.0 %	565	89542	2.4 %	575	88654	1.4 %	554
52	20	9	3	22457	4.7 %	264	21457	0.0 %	265	21743	1.3 %	270	22146	3.2 %	265
53	50	9	4	56875	2.5 %	592	55478	0.0 %	504	57865	4.3 %	590	56478	1.8 %	588
54	100	9	5	165111	0.0 %	915	165724	0.4 %	951	166458	0.8 %	955	169487	2.7 %	971
55	20	5	3	19012	3.7 %	22	18338	0.0 %	24	19208	4.7 %	19	18457	0.6 %	23
56	50	5	4	52836	2.7 %	123	51427	0.0 %	162	51633	0.4 %	138	52411	1.9 %	195
57	100	5	5	153687	0.0 %	309	154215	0.3 %	345	155478	1.2 %	315	154215	0.3 %	323
58	20	7	3	17456	3.4 %	135	16989	0.7 %	124	16875	0.0 %	132	17001	0.7 %	142
59	50	7	4	43875	4.1 %	244	42154	0.0 %	248	43548	3.3 %	264	44123	4.7 %	284
60	100	7	5	120992	0.9	514	119875	0.0	585	121471	1.3	574	125486	4.7	596

					%			%			%			%	
61	20	9	3	22461	4.3 %	212	22145	2.8 %	204	21545	0.0 %	240	22314	3.6 %	240
62	50	9	4	46174	2.0 %	418	45285	0.0 %	456	46532	2.8 %	435	45875	1.3 %	452
63	100	9	5	109886	0.9 %	848	110245	1.3 %	887	108856	0.0 %	924	109206	0.3 %	945
64	20	5	3	16194	5.7 %	18	15324	0.0 %	16	15745	2.7 %	19	15648	2.1 %	14
65	50	5	4	41318	4.1 %	128	39827	0.4 %	132	39678	0.0 %	145	40125	1.1 %	152
66	100	5	5	82547	0.0 %	374	84752	2.7 %	358	86598	4.9 %	395	87542	6.1 %	412
67	20	7	3	18492	4.2 %	139	17739	0.0 %	142	18235	2.8 %	136	18088	2.0 %	139
68	50	7	4	59343	4.3 %	276	56879	0.0 %	306	58689	3.2 %	297	57456	1.0 %	258
69	100	7	5	152487	2.6 %	565	148963	0.3 %	556	150214	1.1 %	587	148562	0.0 %	535
70	20	9	3	19443	3.7 %	228	18756	0.0 %	225	18754	0.0 %	226	18956	1.1 %	225
71	50	9	4	65333	1.7 %	474	64510	0.4 %	468	64235	0.0 %	474	68754	7.0 %	465
72	100	9	5	121454	1.1 %	938	120124	0.0 %	945	122254	1.8 %	954	123254	2.6 %	965
73	20	5	3	15185	4.2 %	13	14568	0.0 %	14	14572	0.0 %	14	15342	5.3 %	16
74	50	5	4	38654	5.7 %	184	36584	0.0 %	198	37684	3.0 %	186	37568	2.7 %	196
75	100	5	5	106163	5.9 %	345	101245	1.0 %	332	103548	3.3 %	332	100245	0.0 %	324
76	20	7	3	14025	1.1 %	123	13873	0.0 %	124	14256	2.8 %	121	14668	5.7 %	125
77	50	7	4	39587	0.0 %	255	40125	1.4 %	252	42154	6.5 %	268	40387	2.0 %	286
78	100	7	5	108654	1.9 %	510	106866	0.3 %	532	106582	0.0 %	563	110240	3.4 %	614
79	20	9	3	22455	1.9 %	242	22047	0.0 %	248	22455	1.9 %	242	22456	1.9 %	245
80	50	9	4	63969	2.3 %	425	62531	0.0 %	435	64578	3.3 %	442	64057	2.4 %	458
81	100	9	5	148472	3.8 %	986	143024	0.0 %	925	149685	4.7 %	954	143257	0.2 %	976
82	20	5	3	17100	4.6 %	28	16354	0.0 %	31	17240	5.4 %	27	16587	1.4 %	25
83	50	5	4	44245	1.5 %	152	43621	0.1 %	150	43578	0.0 %	168	44228	1.5 %	136
84	100	5	5	133645	3.8 %	370	128803	0.0 %	370	129867	0.8 %	396	130454	1.3 %	404
85	20	7	3	16304	0.0 %	94	16476	1.1 %	92	16584	1.7 %	96	17104	4.9 %	110
86	50	7	4	54114	7.7 %	247	50254	0.0 %	253	53212	5.9 %	275	51427	2.3 %	265
87	100	7	5	109655	4.0 %	532	110422	4.7 %	568	108666	3.0 %	654	105487	0.0 %	602
88	20	9	3	15247	1.5 %	172	15027	0.0 %	166	15784	5.0 %	196	15227	1.3 %	196
89	50	9	4	46875	6.4 %	412	44124	0.2 %	425	45732	3.8 %	412	44055	0.0 %	435
90	100	9	5	122179	1.2 %	876	120704	0.0 %	902	121042	0.3 %	956	122335	1.4 %	967
91	20	5	3	11476	2.5	17	11200	0.0	12	11447	2.2	16	11347	1.3	19

					%			%			%			%	
92	50	5	4	39127	1.2 %	141	38654	0.0 %	123	41247	6.7 %	154	39754	2.8 %	112
93	100	5	5	91300	0.8 %	364	90993	0.5 %	358	90554	0.0 %	348	93655	3.4 %	386
94	20	7	3	24580	3.7 %	126	23713	0.0 %	140	25477	7.4 %	128	24875	4.9 %	132
95	50	7	4	57899	5.7 %	188	54755	0.0 %	192	58668	7.1 %	196	57854	5.7 %	192
96	100	7	5	118465	0.0 %	577	121457	2.5 %	556	119586	0.9 %	525	123547	4.3 %	572
97	20	9	3	14131	4.3 %	236	13548	0.0 %	225	14102	4.1 %	222	13547	0.0 %	230
98	50	9	4	50036	0.7 %	469	49868	0.4 %	458	51457	3.6 %	475	49669	0.0 %	468
99	100	9	5	142154	1.4 %	948	140257	0.0 %	932	144755	3.2 %	954	146587	4.5 %	969
100	20	5	5	13365	7.4 %	20	12445	0.0 %	22	12668	1.8 %	19	12547	0.8 %	19
Average				–	2.4	331.6	–	0.5	336.5	–	2.2	343.3	–	2.1	352.1

Regarding Table 7, the GA algorithm outperforms the other algorithms with respect to the average RPE. Furthermore, there is a significant difference between the efficiency of the algorithms. Also, the GA can achieve the best solution among the algorithms for 56 out of 100 test problems.

In order to verify the statistical validity of the results shown in Table 7 and to confirm which algorithm has better performance, a two-sample t-test is conducted in 95% confidence interval. Note that the statistical hypothesis, which is considered in this research, is as: $\begin{cases} H_0: \mu_1 - \mu_2 = 0 \\ H_1: \mu_1 - \mu_2 \neq 0 \end{cases}$. The obtained results are summarized in Table 8.

Table 7
P-Value of the two sample t-test for the proposed algorithms

	SA	**GA**	**PSO**	**PSO-SA**
SA	---	0.000	0.037	0.146
GA	---	---	0.000	0.000
PSO	---	---	---	0.605
PSO-SA	---	---	---	---

Regarding Table 8, the GA outperforms other proposed algorithms with respect to the $P - Value < 0.05$. Also, other algorithms have the same performance in the confidence interval of 0.95. The means plot and least significant difference (LSD) intervals (at the 95 % confidence level) are depicted in Figure 12 for four algorithms.

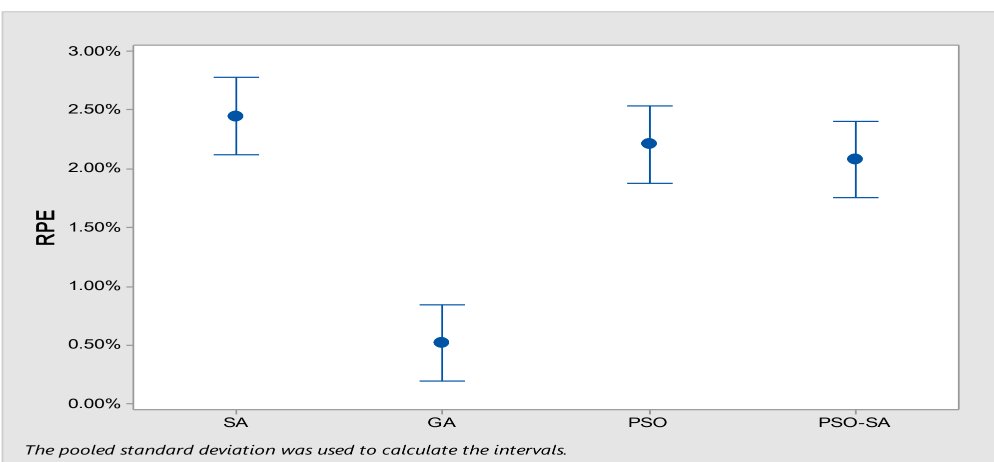

Fig. 12. The LSD plot of the proposed algorithms

5.3. Analysis of parameters of test problems

Analysis of the number of jobs: in order to investigate the effect of number of jobs on the proposed algorithms, the interaction between the algorithms and number of jobs is illustrated in Figure 13. As can be seen, in the entire cases, the GA has better performance.

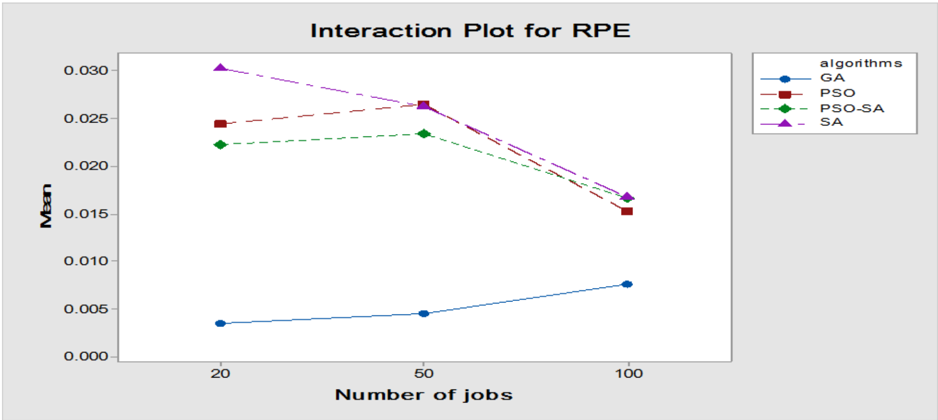

Fig. 13. Plot of average RPE for the interaction between the algorithms and number of jobs

Analysis *of the number of stages:* another plot for interaction between the algorithms and number of stages is depicted in Figure 14. Considering Figure 14, the GA works better than other algorithms in all cases.

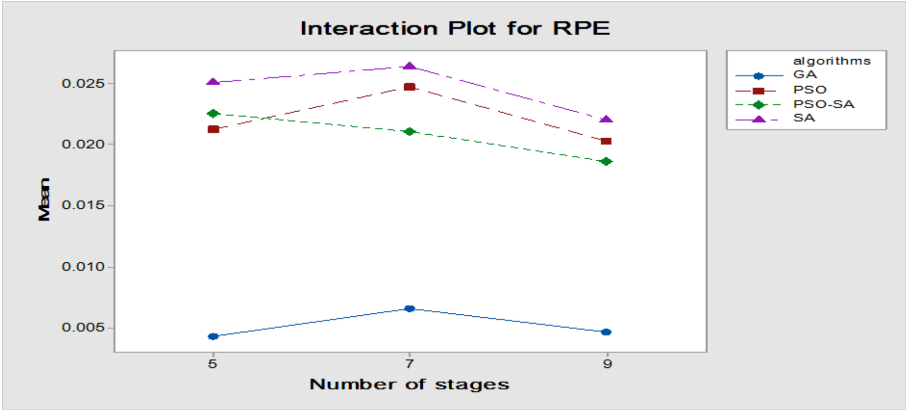

Fig. 14. Plot of average RPE for the interaction between the algorithms and number of stages

Analysis of the number of subcontractors: finally, Figure 15 shows the interaction plot between number of subcontractors and quality of the algorithms. Regarding to Figure 15, the GA works better than the PSO, SA, and PSO-SA.

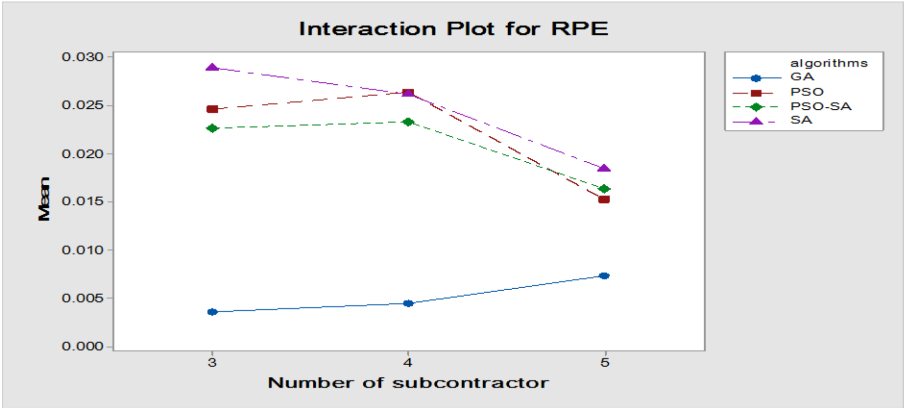

Fig. 15. Plot of average RPE for the interaction between the algorithms and number of subcontractors

6. Conclusion

Flexible Flow Shop is a common manufacturing system in which a set of *n* jobs are processed on *s* stages. This paper aims to consider outsourcing options in a flexible flow shop scheduling problem with cost-related objective functions. In this paper, we developed a mathematical model for the research problem. Regarding the NP-hardness of the research problem, we used metaheuristic algorithms, SA, GA, PSO and PSO-SA for solving problems with medium to large-size test problems. The obtained results demonstrated that the GA has better performance.

Future research may consider outsourcing for other scheduling environments or consider some assumptions such as release date for jobs or consider the batch shipment constraint for the outsourced jobs. Proposing other heuristic and metaheuristic approaches, considering other objective functions, and using data from the real case study can be other clues of future researches.

Acknowledgment

The second author would like to thank the financial support of the Babol Noshirvani University of Technology for this research under grant number BNUT/388026/98.

References

Ahmadizar, F. & Amiri, Z. (2018). Outsourcing and scheduling for a two-machine flow shop with release times. *Engineering Optimization*, 50(3), 483-498.

Asadi-Gangraj, E. (2018). Lagrangian relaxation approach to minimizing makespan in hybrid flow shop scheduling problem with unrelated parallel machines. *Scientia Iranica E*, 25(6), 3765-3775.

Asadi-Gangraj, E., (2017). Heuristic Approach to Solve Hybrid Flow-Shop Scheduling Problem with Unrelated Parallel Machines. *International Journal of Industrial Engineering & Production Research*, 28(1), 61-74.

Behnamian, J., (2020). Parallel Jobs Scheduling with a Specific Due Date: a Semi-Definite Relaxation-Based Algorithm. *Journal of Optimization in Industrial Engineering*, 13(2), 199-210.

Chan, F.T., Kumar, V. & Tiwari, M.K. (2009). The relevance of outsourcing and leagile strategies in performance optimization of an integrated process planning and scheduling model. *International Journal of Production Research*, 47(1), 119-142.

Chen, Z.-L. & Li, C.-L. (2008). Scheduling with subcontracting options. *IIE transactions*, 40(12), 1171-1184.

Choi, B.-C. & Chung, J. (2011). Two-machine flow shop scheduling problem with an outsourcing option. *European Journal of Operational Research*, 213(1), 66-72.

Choi, B.-C. & Chung, K. (2016). Min–max regret version of a scheduling problem with outsourcing decisions under processing time uncertainty. *European Journal of Operational Research*, 252(2), 367-375.

Chung, D., Lee, K., Shin, K. & Park, J. (2005). A new approach to job shop scheduling problems with due date constraints considering operation subcontracts. *International Journal of Production Economics*, 98(2), 238-250.

Chung, D.-Y. & Choi, B.-C. (2013). Outsourcing and scheduling for two-machine ordered flow shop scheduling problems. *European Journal of Operational Research*, 226(1), 46-52.

Eberhart, R. & Kennedy, J. (1995). A new optimizer using particle swarm theory. In: Micro Machine and Human Science. *Proceedings of the Sixth International Symposium*, 39-43. IEEE

Goldberg, D.E. & Holland, J.H. (1988). Genetic algorithms and machine learning. *Machine learning*, 3(2), 95-99.

Guo, X. & Lei, D. (2014). Bi-objective job shop scheduling with outsourcing options. *International Journal of Production Research*, 52(13), 3832-3841.

Haoues, M., Dahane, M., Mouss, N.K. & Rezg, N. (2013). Integrated Optimisation of In-House Production and Outsourcing Strategy: Genetic Algorithm Based Approach. *IFAC Proceedings Volumes*, 46(7), 420-425.

Hosseini, S.M.H. (2019). Modeling and Solving the Job Shop Scheduling Problem Followed by an Assembly Stage Considering Maintenance Operations and Access Restrictions to Machines. *Journal of Optimization in Industrial Engineering*, 12(1), 63-78.

Johnson, S.M. (1954). Optimal two- and three- stage production schedules with setup times included. *Naval Research Logistics* (NRL), 1(1), 61-68.

Lee, I.S. & Sung, C. (2008). Minimizing due date related measures for a single machine scheduling problem with outsourcing allowed. *European Journal of Operational Research*, 186(3), 931-952.

Lee, I.S. & Sung, C. (2008). Single machine scheduling with outsourcing allowed. *International Journal of Production Economics*, 111(2), 623-634.

Lee, K. & Choi, B.-C. (2011). Two-stage production scheduling with an outsourcing option. *European Journal of Operational Research*, 213(3), 489-497.

Lee, Y.H., Jeong, C.S. & Moon, C. (2002). Advanced planning and scheduling with outsourcing in manufacturing supply chain. *Computers & Industrial Engineering*, 43(1-2), 351-374.

Lei, D. & Guo, X. (2016). A shuffled frog-leaping algorithm for job shop scheduling with outsourcing options. *International Journal of Production Research*, 54(16), 4793-4804.

Mishra, N., Choudhary, A. & Tiwari, M. (2008). Modeling the planning and scheduling across the outsourcing supply chain: a Chaos-based fast Tabu–SA approach. *International Journal of Production Research*, 46(13), 3683-3715.

Moghaddam, A., Yalaoui, F. & Amodeo, L. (2012). A Genetic-based Algorithm to Find Better Estimation of Non-Dominated Solutions for a Bi-objective Re-entrant Flowshop Scheduling Problem with Outsourcing. *IFAC Proceedings Volumes*, 45(6), 1383-1388.

Mokhtari, H. & Abadi, I.N.K. (2013). Scheduling with an outsourcing option on both manufacturer and subcontractors. *Computers & Operations Research*, 40(5), 1234-1242.

Mokhtari, H., Abadi, I.N.K. & Amin-Naseri, M.R. (2012). Production scheduling with outsourcing scenarios: a mixed integer programming and efficient solution procedure. *International Journal of Production Research*, 50(19), 5372-5395.

Nahavandi, N. & Asadi-Gangraj, E. (2014). A new lower bound for flexible flow Shop Problem with unrelated parallel machines. *International Journal of Industrial Engineering & Production Research*, 25(1), 65-70.

Nayeri, S., Asadi-Gangraj, E. & Emami, S. (2019). Metaheuristic algorithms to allocate and schedule of the rescue units in the natural disaster with fatigue effect. *Neural Computing and Applications*, 31, 7517–7537.

Neto, R.F.T., Godinho Filho, M. & Da Silva, F.M. (2015). An ant colony optimization approach for the parallel machine scheduling problem with outsourcing allowed. *Journal of Intelligent Manufacturing*, 26(3), 527-538.

Neto, R.T. & Godinho Filho, M. (2011). An ant colony optimization approach to a permutational flowshop scheduling problem with outsourcing allowed. *Computers & Operations Research*, 38(9), 1286-1293.

Qi, X. (2008). Coordinated logistics scheduling for in-house production and outsourcing. *IEEE Transactions on Automation Science and Engineering*, 5(1), 188-192.

Qi, X. (2009). Two-stage production scheduling with an option of outsourcing from a remote supplier. *Journal of Systems Science and Systems Engineering*, 18(1), 1-15.

Qi, X. (2011). Outsourcing and production scheduling for a two-stage flow shop. *International Journal of Production Economics*, 129(1), 43-50.

Rezaeian, J. & Zarook, Y. (2018). An efficient bi-objective genetic algorithm for the single batch- processing machine scheduling problem with sequence-dependent family setup time and non-identical job sizes. *Journal of Optimization in Industrial Engineering*, 11(2), 65-78.

Ruiz-Torres, A.J., Ho, J.C. & López, F.J. (2006). Generating Pareto schedules with outsource and internal parallel resources. *International Journal of Production Economics*, 103(2), 810-825.

Tirkolaee, E.B., Goli, A., & Weber, G.W. (2020). Fuzzy mathematical programming and self-adaptive artificial fish swarm algorithm for just-in-time energy-aware flow shop scheduling problem with outsourcing option. *IEEE transactions on fuzzy systems*, DOI: 10.1109/TFUZZ.2020.2998174.

Wang, S. & Cui, W. (2020). Approximation algorithms for the min-max regret identical parallel machine scheduling problem with outsourcing and uncertain processing time, *International Journal of Production Research*, Doi: 10.1080/00207543.2020.1766721.

Zhong, W. & Huo, Z. (2013). Single machine scheduling problems with subcontracting options. *Journal of Combinatorial Optimization*, 26(3), 489-498.

Enayati, M., Asadi-Gangraj, E., Paydar, M. (2021). Scheduling on flexible flow shop with cost-related objective function considering outsourcing options.
Journal of Optimization in Industrial Engineering, 14(2),53-72.

http://www.qjie.ir/article_677014.html
DOI:10.22094/JOIE.2020.1873983.1674